

CHAPTER 6

COMPUTER MEMORIES

INTRODUCTION

The memory of a computer holds (stores) program instructions (what to do), data (information), operands (affected, manipulated, or operated upon data), and calculations (ALU results). The CPU controls the information stored in memory. Information is fetched, manipulated (under program control) and/or written (or written back) into memory for immediate or later use. The internal memory of a computer is also referred to as main memory, global memory, main storage, or primary storage. Do not confuse it with secondary or auxiliary memory (also called mass storage) provided by various peripheral devices. In newer computers you also will encounter a number of small and independent local memories that are used for a variety of purposes by embedded microprocessors. You have already learned about cache memory that lies between the CPU and main memory.

After completing this chapter, you should be able to:

- Describe the organization of memory
 - Describe the operation of main memory
 - Recognize the types of memory and describe how they function
-

TOPIC 1—MEMORY ORGANIZATION AND OPERATION

The main memory of a computer is used for storing programs, data, calculations, and operands. Memory is used in all types of computer systems including mainframes, minicomputers, and microcomputers. The amount of main memory each type of computer has varies according to the configuration. A wide variety of memory types is being used. To simplify our discussion, we have divided memory into two general categories: read/write (random access) memory and read-only memory. Within the read/write group, we discuss magnetic (core and film) memories and semiconductor (static and dynamic) memories. Read-only memory can be subdivided into factory programmed parts called read-only memory (ROM) and user programmable devices called programmable read-only memory (PROM). This classification system is illustrated in figure 6-1. Let's take a look at some of the terminology used with regard to the computer's memory.

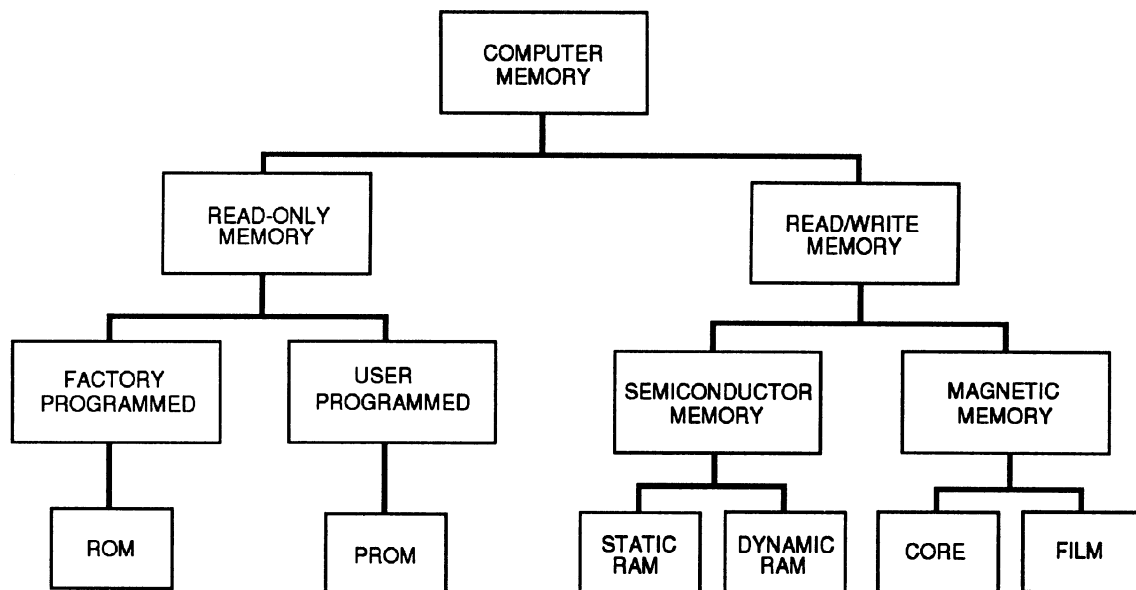
TERMINOLOGY

The following terms need to be explained at this point:

- **Memory** —Memory generally refers to the actual hardware where the programs, data, calculations, or operands are stored.

- **Memory address** —A memory address is a particular location of a larger memory array. Usually one memory address contains one word of data. A word is one packet of information for the computer and is usually composed of many bits. Computers exist that use 1-bit words, 8-bit words, 16-bit words, 32-bit words, and 64-bit words. Handling computer data in 8-bit words is so common that the 8-bit word has its own name, the byte. Half of a byte is called a nibble (4 bits).

- **Capacity (memory size)** —Capacity is an important aspect of system performance; it is a useful and convenient way to describe the size of memory. At the individual part level, a computer's memory may be



35NVM017

Figure 6-1.—Classification system of memory.

described as containing 65,536 bits; or, alternately, it may be called an “8K by 8” memory. Most computer memory sizes are described as a specific number of words. It is assumed that the word size is geared to the particular computer that is used with the memory. Computer memory sizes are given in **K increments**, or roughly 1,000 word blocks. The exact size of a 1K block is 1,024, which is 2^{10} .

- **Access time** —Access time is a measure of the time required to read from or write the data to a particular address in the memory. It is the interval from the instant at which a request for data is initiated until the data is available for use. It can range from a few nanoseconds (ns) to microseconds (μs).

- **Destructive readout** —When data is read from memory, the stored data is extracted (removed) from memory and in the process the data is erased in the source. Because the data is lost, the process is referred to as destructive readout. If it is desired to restore the same data at the same storage location, the word must be rewritten after reading. Read/write memory such as a core memory is an example of destructive readout.

- **Non-destructive readout** —If the data in a memory is not destroyed in the reading process, the system has non-destructive readout. This means the data can be read over and over again without being rewritten. A flip-flop is an example of nondestructive readout. Sensing the output voltage (reading) from a given side of a flip-flop generally does not change the state of the flip-flop and the stored data is retained.

- **Volatile memories** —Volatile memories are memories that lose their contents when the **power** is turned off. A semiconductor memory is an example.

- **Nonvolatile memories** —Nonvolatile memories are memories that do **not** lose their contents when power is removed. Core memory is an example.

MEMORY ORGANIZATION

Memory organization is two-fold. First we discuss the hardware (physical) organization, then the internal architecture. The type of computer and its size do not reflect the type of memories that the computer uses. Some computers have a mixture of memory types. For example, they may use some type of magnetic memory (core or film) and also a semiconductor memory (static or dynamic). They also have a read-only memory which is usually a part of the CPU.

Memory in a computer can vary from one or more **modules** to one or more **pcb's**, depending on the computer type. The larger mainframe computers use the modular arrangement, multiple modules (four or more), to make up their memories. Whereas, minicomputers and microcomputers use chassis or assemblies, cages or racks, and motherboard or backplane arrangements. Minis and micros use multiple components on one pcb or groups of pcb's to form the memory.

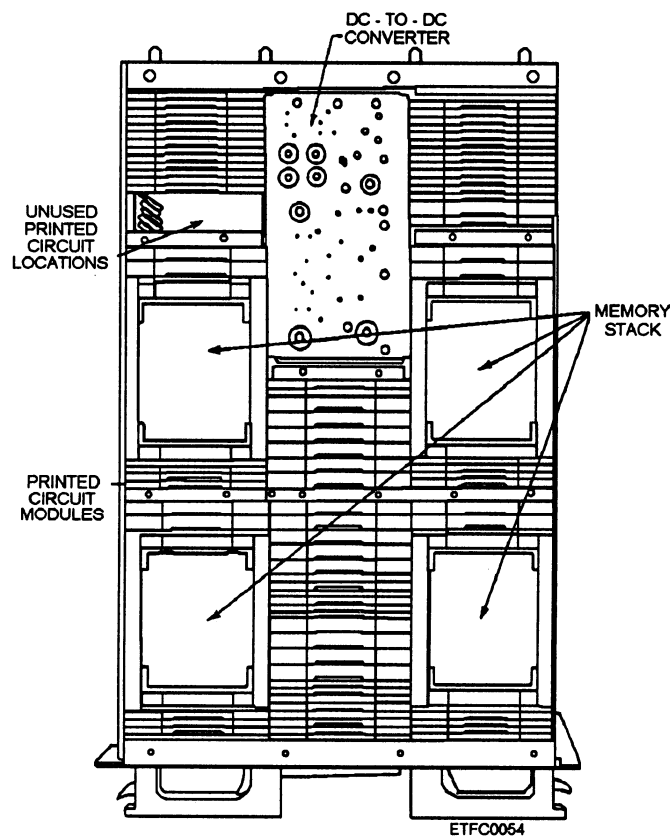


Figure 6-2.—Memory module/unit.

Memory Modules

Memory modules are made up of multiple pcb's (support circuitry) and memory components (stacks [core or film] or semiconductor pcb's with support circuitry) to form one memory **module** or **unit**. Figure 6-2 is an illustration of a large memory module; one of four to a single computer set. Memory modules are interchangeable with other modules of the same type and size in the same computer set. Each module provides a fixed number of memory words with a fixed number of bit positions for each word. Some memory modules are designed with the capability to receive requests from more than one central processing unit or I/O section. These **multiported memory modules** process memory requests on a priority basis. While the module is processing a request from one section, the remaining possible requesters are locked out, so to speak, by the module logic until the completion of the pending request. If two requests are received simultaneously, then the highest priority requester is cycled first. Memory modules may contain magnetic or semiconductor memory types. Some computers use both but in different modules. The size of memory in terms of bits and arrangement contained on each of these types depends on the requirements and design of

the computer. Consult your technical manual for the exact size and arrangement.

Memory Pcb's

Computers that use a small number of pcb's as their memories are usually of the semiconductor type. In mainframe semiconductor memory, pcb's and support circuitry are contained in a module or unit. In minicomputers and microcomputers, memory can be contained on as few as one pcb, or as many as half-a-dozen pcb's. When there is more than one pcb, they are usually arranged in a group together in the computer's frame or cabinet. Micros can also use a bank of IC chips for their memories. The IC chips are mounted on single inline memory modules (SIMMs) (fig. 6-3), single inline packages (SIPs), or single inline

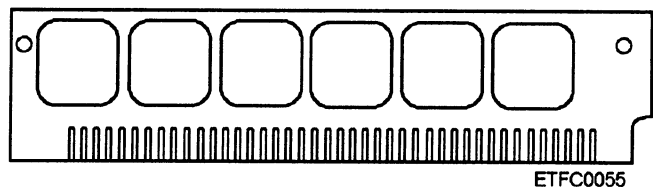


Figure 6-3.—Illustration of a 30-pin single inline memory module (SIMM).

pin packages (SIPPs). Memory pcb's also operate on a request basis, but unlike memory modules, there is not a priority sequence to go through. The request is made by requestor, the control circuitry selects either a read or a write operation, and the timing circuitry initiates the read and/or write operations.

Memory Architecture

The memory architecture, regardless of the memory type, is consistent. Memories are typically organized in square form so they have an equal number of rows (x) and columns (y) (fig. 6-4). Each intersection of a row and column comprises a **memory word address**. Each memory address contains a **memory word**. The selected memory address can contain one or more bits. But for speed and practicality, for a given computer design, the word size typically relates to the CPU and is usually the size of its registers in bits. Word sizes typically range in increments of 8, 16, 32, or 64 bits. Figure 6-5 represents an address with an 8-bit word. The methods used for the arrangement of the rows and columns vary in a given type of memory. The rows and columns are arranged in arrays, memory planes, or matrices.

MEMORY OPERATIONS

Memories operate on a request, selection, and initiate basis. A memory request or selection and a memory word location are transmitted from the

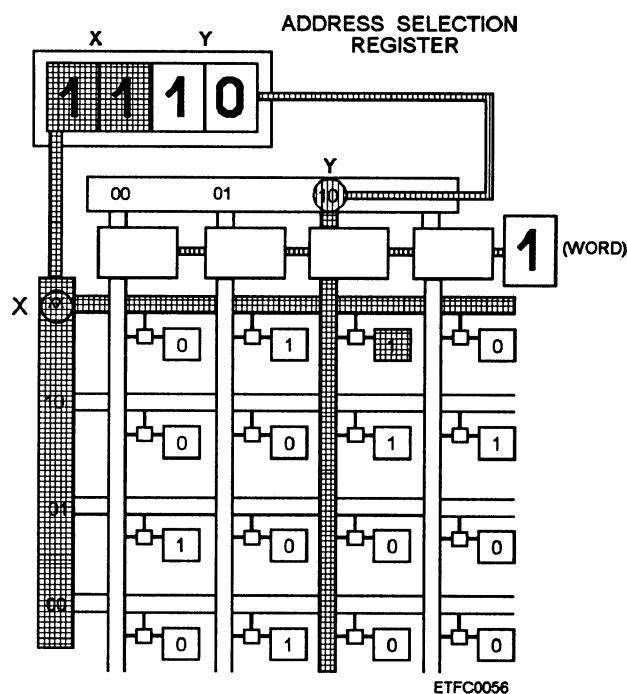


Figure 6-4.—Row (X), column (Y) organization.

requestor (CPU or I/O sections) to the memory section. The computer's internal bus system transmits the memory request or selection and location to the memory section. The memory operations, regardless of the computer type, share some basic commonalities. Key events must occur to access and store data in memory. Some items only occur with certain types of memories, and we discuss these as you study each different type of memory. We also discuss the items that are common to most memories: **control circuits**, **timing circuits**, and **memory cycle**. In addition, we present methods used for **detecting faults** and **protecting memory**.

Memory Interface Circuits

The memory interface circuits include all the lines of communication (buses) and the interfacing register between the requestor (CPU or I/O(C)) and memory. The communications lines include some of the following:

- Data (bidirectional bus)
- Control lines (write byte and interleave [for large computers])
- Memory request
- Read and write enables
- Data ready
- Data available

The **interface (data) register** (often designated as the "Z" register) functions as the primary interfacing component of memory. Before the read/write operation, this register transfers the selected memory address to the **address register**. All data entering and leaving the memory is temporarily held in this data register. In a write operation, this register receives data from the requestor; and in a read operation, this register transmits data to the requestor. For computers with destructive readout, it routes the data back to memory to be rewritten.

Control Circuits

The control circuits set up the signals necessary to control the flow of data and address words in and out of memory. They screen the request or selection by units external to memory—the CPU and/or IO(C). Depending on the computer type, some of the more common uses of the control circuits include:

Memory Cycle

Main memory is a read/write memory that allows data to be retrieved (read) and stored (written) in what is known as the **memory cycle**. The memory cycle includes reading the data out of memory and/or writing the data into memory, either by a read/write operation or by separate read and write operations. The memory cycle is based on fixed (constant) time periods for reading and/or writing data from and into memory. As soon as read and/or write operations are initiated, almost simultaneously, **address translation** occurs, then the **read** and/or **write** cycle or cycles begin.

MEMORY ADDRESS TRANSLATION.— One of the most important processes that must occur before a read or write cycle begins is the memory address translation. Look at memory as a sequence of memory locations starting at address 0 and extending to the maximum memory address available to the requester (CPU or IO(C)). Receiving or sending mail uses a similar concept. Before mail can be received or sent, there must be an address on the envelope. Memory uses the same principle. Memory logic identifies the memory address where a memory word is to be read from or **written** into in memory. A memory address can be anyone of the entire range of memory addresses (0 to maximum). To identify the desired memory address, the memory logic uses a register designated as the address register and/or translators or decoders. The memory logic receives the logical address from the CPU or I/O and temporarily stores it in the address register, and then converts it to a physical address that can be read from or written into.

Memory Address Register and/or Translator (Decoder).— The address register and/or translator identifies the exact location from which to read the bits or write them. The contents of the address register or translator identifies the memory address. The memory logic is designed to make its selection based on the type of memory it uses. It can be designed to identify a memory address of a single memory pcb or it can be designed to identify an address located in one of four or more memory modules.

Memory Address Word.— As stated in the architecture of memory, the word contained in the memory address can be one or more bits, most computers have words with at least 8 bits and some up to 128 bits in length. Variations of reading and/or writing from and to memory can include the upper or lower half of the word, or any other variation within the design of a given computer. Variations are based on the

instruction types and the program. Also, if a computer is identified as an 8-bit computer and a 16-bit word is required for a read or write operation, then two consecutive memory addresses would have to be used to complete the operation. There are many other variations; the instruction repertoire set of your computer and the technical manual will provide details of your computer's memory operations and limitations.

Here are two examples of memory address translation. For the first example, use figure 6-6 as a reference; it shows a 4-bit memory address. The memory address register or translator contains 16, as follows:

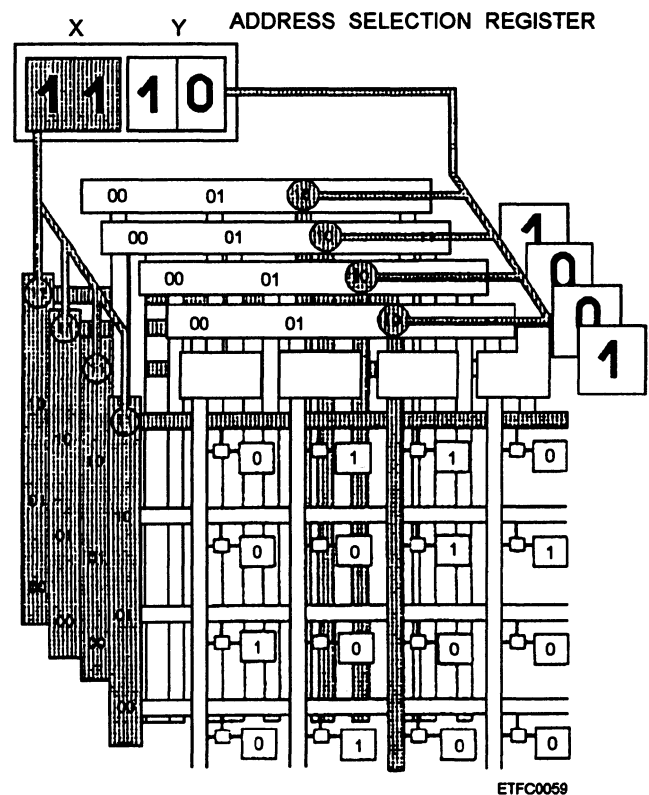
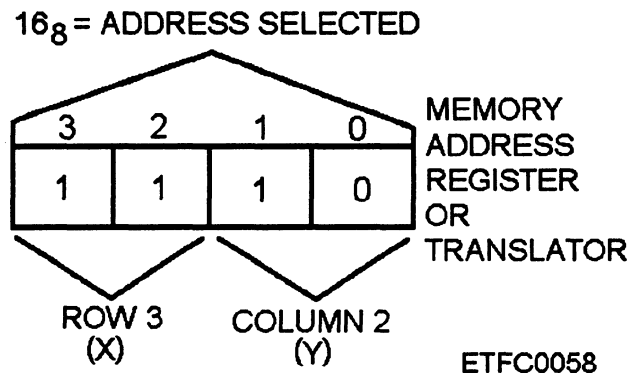


Figure 6-6.—4-bit address.

For our second example, refer to figure 6-7. It shows an 8-bit memory address, a memory module decoder, and four memory modules (each with 100₈ addresses). In the figure, the memory address register or translator contains 372₈. When a memory reference takes place, the address translation logic decodes the two most significant bits of the 8-bit logical address to determine and select the applicable memory module. The lower six bits of the logical address are passed to all the memory modules to determine the row (x) and column (y) intersection, but only the selected memory module decodes the address. In this example, memory address 72₈ of memory module 3 is addressed. This means a word will be read from (or written into) this address. The following is a breakdown of the address:

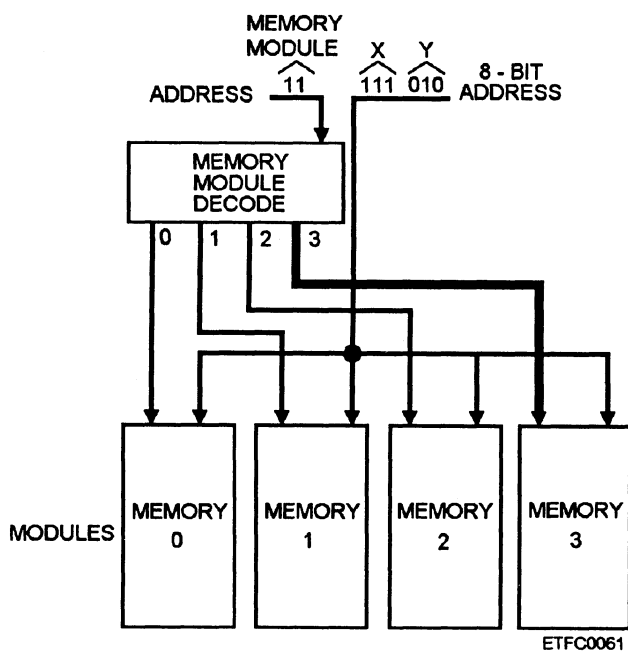
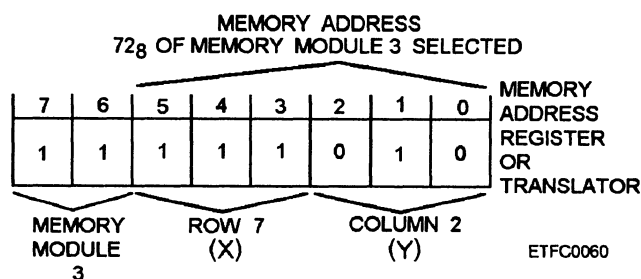


Figure 6-7.—8-bit address with memory module.

READ CYCLE.— Processing a read request requires memory to read data from the addressed memory location and transmit the data via a bus to the requesting section of the computer where it is used for calculations or output to another device. The information read from memory can be part of a program, general data, calculations, or operands.

Remember, information read from a destructive readout memory has to be written back into memory or it will be lost. Also remember, for some memory types, read and write operations are separate. Depending on the computer's instruction set, the information can be read from anywhere in memory or any part of a memory address.

WRITE CYCLE.— A write request, on the other hand, causes memory to accept information from a bus and to store (write) the information in the addressed memory location. Again the information can be part of a program, data, calculations, or an operand. The information can come from the CPU or another device. For those memories that are destructive, the write cycle is a must to retain the original data after a read. Otherwise, it is a separate operation. Just as in a read cycle, the computer's instruction set allows the information to be written into any memory address or part of a memory address in a read/write memory.

INTERLEAVE.— A large memory may be organized in several modules, each covering a portion of the addressable space. The effective speed of this memory can be increased if memory access cycles in different modules are overlapped. In such a system, the CPU requests a read or write operation in one module. Then, it requests other memory operations in other modules before the result from the first module becomes available. Because program instructions are usually fetched from successive locations in the main memory, overlapped operation may be achieved by arranging memory addresses such that successive addresses refer to different modules. For example, if there are four modules, the first module should contain words 0, 4, 8, . . . etc; the second module words 1, 5, 9, . . . etc; and so on. The increase in speed with memory interleaving is achieved at the expense of increased complexity in the CPU and memory control circuitry.

Memory Fault Detection

A variety of methods is used to ensure the accuracy of data written into and read from the memory section. The methods include parity check and error bit detection and/or correction.

PARITY CHECK.— Parity check is one of the simplest methods used to detect read/write errors in core memory. The strategy is simple; the computer counts the number of ones in a memory word, then adds an extra bit to make the total number of ones either an even number or an odd number depending on whether the

computer uses even or odd parity. The process relies on the exclusive-OR operation to count the ones. Parity checks are designed to identify the loss (1 to 0) or gain (0 to 1) of a single bit during the read/write process. When the data is read from memory, it is checked for an even number of bits for even parity computers or an odd number of bits for odd parity computers. A difference causes the generation of a **parity error signal** or other type of error to the requestor. If no error condition exists, the parity bit is dropped and the computer continues processing. Parity checks do not provide for correction of the error condition.

ERROR BIT DETECTION AND/OR CORRECTION.—Newer computer designs use error detection and correction circuitry for their semiconductor memories, modules, or pcb's. The error detection and correction circuits allow for the detection and correction of single bit errors and the detection of double and sometimes 3-bit errors during read/write operations.

The error detection and correction circuits use a **Hamming code** to identify the configuration of ones and zeros stored in a particular memory location or group of bits. Additional storage for check bits is required for each memory address. The number of check bits varies with the number of data bits being tested. For instance, six check bits are used for a 16-bit data word. The check bits are generated by the error detection and correction circuits during the write operation and are written into the memory address with the data.

During read operations, the stored check bits are compared with the error detection and correction generated check bits of the data read. Differences in the check bit patterns can be used to correct single bit data errors and at least identify the presence of double bit or greater errors. The error detection and correction circuitry will indicate the detection of any error to the CPU. In computers with the error detection and correction capability, the correction circuits can be enabled or disabled by CPU instruction. The error detection circuits, however, function at all times.

Memory Protection

Many computers provide controlled access to specified segments of memory through the use of memory protection registers. The **memory protection register set** (usually three registers) is used to restrict read/write operations in the protected area. In one form of the memory protection register set, the boundaries of

the protected area are defined by the **memory protect lower limit register**, which contains the lower boundary address and the **memory protect upper limit register**, which contains the upper boundary address. All addresses between the upper and lower limits are protected. The memory protection control register contains three control bits that determine the allowable operations in the protected area. The memory protection control bits are set (1) to allow each of the following three operations (in any combination):

- Read instruction (execute protected)
- Read operand (read protected)
- Write operand (write protected)

After a request has been accepted, the memory protection logic checks the address to determine if it is in the protected area. If the address is within the boundaries, the operation being requested is checked to see if it is allowable. An allowable operation is executed. In the event an attempted operation is not allowed, a **memory protect fault interrupt** is sent to the requestor. Other forms of memory protection registers identify the following:

- Starting address
- Block size (number of addresses)
- Protection function

The basic protection functions are the same for all computers; however, some computers may have an additional control bit to allow indirect addressing within the protected area.

Another form of memory protection called **memory lockout** is used by larger computers to prevent access to particular areas of memory by task state instructions. Memory lockout prevents task state programs (application programs) from accessing segments of main memory reserved for interrupt processing and other executive functions. The lockout feature is disabled when the CPU enters a particular executive or interrupt state and enabled when the CPU enters the task state.

MEMORY TYPES

As stated at the beginning of this topic, we have divided the memory types into two categories: **read/write** and **read-only** memories. You will learn more about these in the next two topics.

TOPIC 2—READ/WRITE MEMORIES

In read/write memories, the data can be retrieved from memory, altered, and written back into memory. This can be done either independent of a write operation or as part of the first half of a read/write operation where the information must be rewritten back into memory to restore the original data. Read/write memories are random access in nature. They are categorized according to the materials they are constructed from; not by their basic operations. Their physical makeup can be **magnetic** or **semiconductor**. Both types have advantages and disadvantages. Semiconductor memories cost less, are faster in terms of storage and access time, and use nondestructive readout. They also require less space for the same number of bits as a magnetic memory. Magnetic memories are relatively low in cost, require less power, and are nonvolatile (they retain the information after the power is removed).

In our discussion of the two types of memory, you will study specifics about their **architecture**, **address selection**, and **read/write cycles**; how address selection and the read/write cycle are performed; and any circuitry that is peculiar to that type of memory. First we discuss two types of magnetic memory (core and film), then semiconductor memory.

Magnetic memories use magnetic material as a means of recording binary data. Basically, a magnetic field is applied to a memory cell (bit); the magnetic field is generated by passing a current through the conductor. Magnetic memory is a **non volatile** form of storage. It retains its magnetic state (direction of flux lines) in the absence of current flow through the conductors on which the core or film is assembled. Only current flow in the opposite direction of sufficient magnitude to overcome the magnetic field of the core or film and to magnetize it in the new direction will change the state of the core or film. Loss of power should not cause loss or the data retained in core or film memory.

The major difference between core memory and film memory technology is the physical structure of the material used. Mated film memory is easier to magnetize, which increases the speed of read/write operations. Also, less power is required for these operations. Mated film memory is also more compact and durable, and twice as many mated film memory cells can be put in the same space as ferrite core memory cells for the same amount of power.

CORE MEMORY

Core memory is used as one of the primary storage media of digital computers. It is used primarily on large mainframes and minicomputers. Depending on the mainframe or minicomputer, core memory is contained in **memory modules**; usually two to four large memory modules to a mainframe computer set or one to four small modules in a minicomputer.

Core Architecture

Magnetic core storage is composed of hundreds of thousands of very small doughnut-shaped **ferrite cores** (fig. 6-8). The ferrite cores are strung together on grids of very thin wires known as **core planes**. Each core can store one binary bit (0 or 1) of data. A core is magnetized by current flow through the wires on which the core is strung. A core magnetized in one direction represents a binary zero, and when magnetized in the opposite direction, a binary one. The direction the core is magnetized is dependent on the direction of current flow through the wires on which it is strung. Figure 6-8 shows the magnetization of a core based on the direction of current flow.

CORE WINDINGS (FOUR-WIRE).— Magnetic cores are strung on several fine wires to allow for the reading and writing of data in core. Two basic methods are used to string cores, the four-wire method and the three-wire method. Core windings strung through each core using the four-wire method consist of 2 drive lines (X and Y), 1 sense line, and 1 inhibit line.

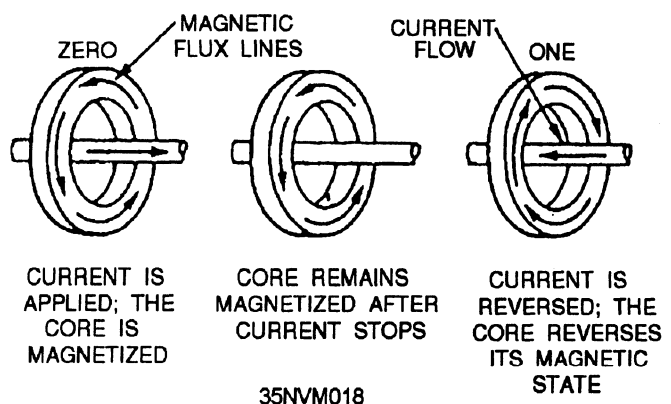


Figure 6-8.—Magnetizing a ferrite core.

An example of a four-wire core with all windings is shown in figure 6-9.

- **Drive lines** —Each drive line provides 1/2 of the current necessary to change the state of the core. In other words, current must flow in the same direction in both drive lines to change the direction of magnetic flux in the core (zero to one or one to zero).

- **Sense line** —The sense line is used when reading data from cores. The sense line detects the change in state of the core from one to zero.

- **Inhibit line** —The inhibit line is used during the write or restoring process. Current flow in the inhibit line opposes or inhibits the drive line currents attempting to change a core from zero to one. Simply put, the inhibit line inhibits writing ones.

THREE-WIRE CORES.— A three-wire core uses a digit line, a word line, and a sense line. The digit and word lines combine to perform the functions of the X and Y drive lines and the inhibit line. The sense line performs the same function as in the four-wire cores.

CORE STORAGE LAYOUT.— As each core can store but one binary bit of data, large numbers of cores are required for effective storage of large amounts of data. Core storage or core memory is designed to store a fixed number of **memory words**. Each core stores

one bit position of one of the memory words. The length of a memory word (number of bit positions) varies from system to system, but common lengths include 8, 16, 32, and 64 bits. The size of core storage, or its **memory capacity**, is determined by the number of memory words that can be used or addressed to store and retrieve data. To accommodate the memory capacity of any size, the memory words are organized into **matrices**.

Matrices.— Magnetic cores are arranged into matrices to simplify addressing, reading, and writing operations. An example of a basic four-wire magnetic core matrix is shown in figure 6-10. Each core in the matrix must have 2 drive lines (X and Y), an inhibit line, and a sense line intersecting through the center of the core. The most common four-wire core matrix is the 64 by 64 **array**. We base our discussion of matrices on this size.

Arrays — A 64 by 64 array contains 64 X drive lines and 64 Y drive lines. By selecting one X drive line and one Y drive line, read and write current can be applied to any one of the 4096 ($64 \times 64 = 4096$) cores in the array. As each drive line contains 1/2 of the read or write current, only the core with full read or write current passing through it will be switched. The inhibit line is threaded in parallel with the X or Y drive lines.

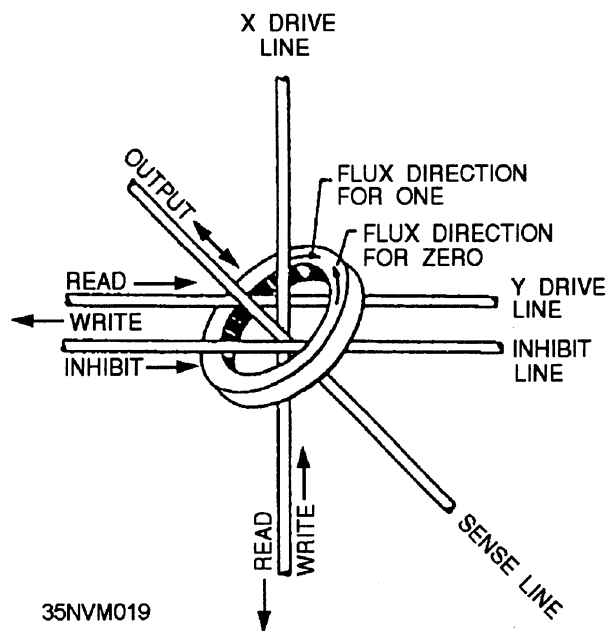


Figure 6-9.—Four-wire magnetic core.

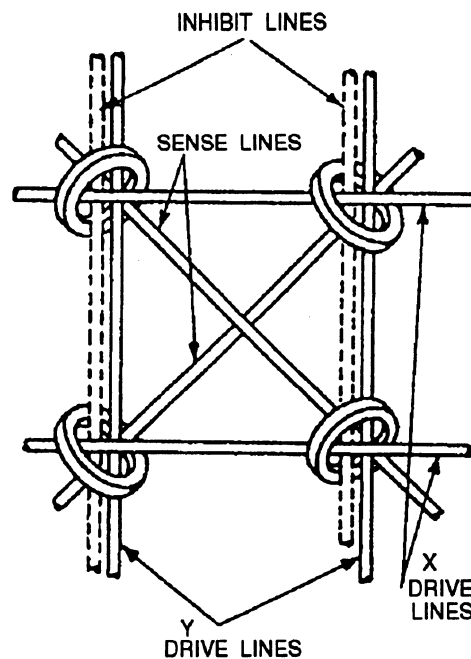


Figure 6-10.—Four-wire magnetic core matrix.

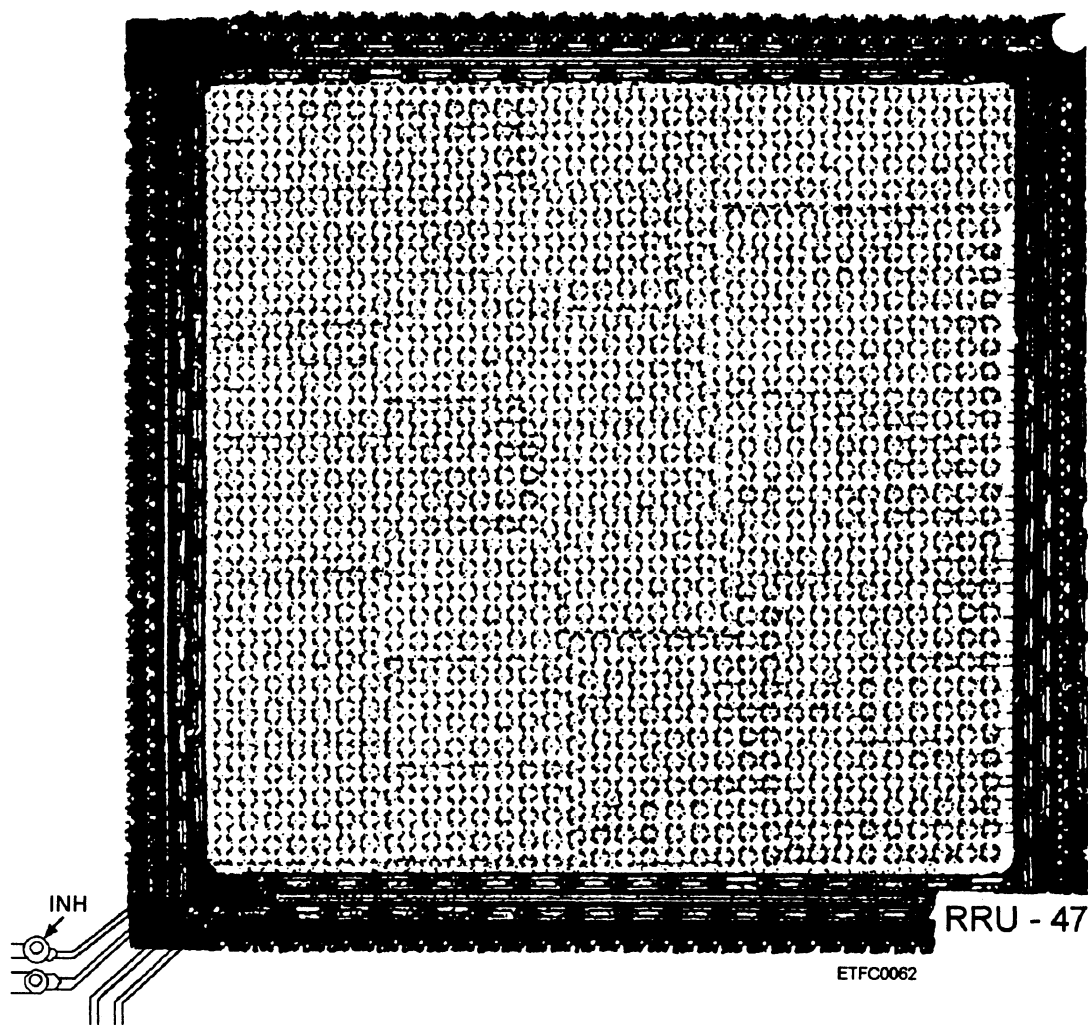
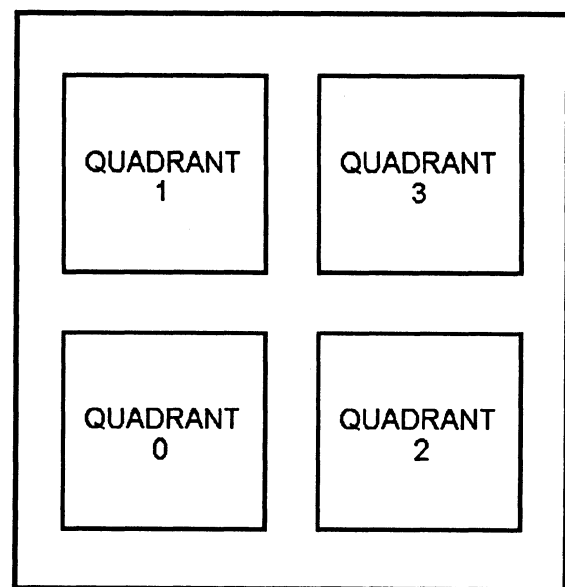


Figure 6-11.—64 × 64 array memory quadrant.

The sense line is threaded through all the cores in the array. One 64 by 64 array forms one quadrant of a **memory plane**. Figure 6-11 is an illustration of a 64 × 64 array.

Memory Plane.—Each quadrant of a memory plane (four in each memory plane) contains one bit position of 4096 memory words. An example of a memory plane is shown in figure 6-12. Each memory plane will provide 4 bit positions (one for each quadrant) over 4096 addresses when using 64 by 64 arrays. The memory plane is the basic building block of the **memory stack**.

Memory Stack.—The memory stack contains all the core of the device and the associated circuitry, which includes the X and Y drive lines, inhibit lines, and sense



ETFC0063

Figure 6-12.—Memory plane.

lines. Figure 6-13 is an illustration of a portion of a core memory plane in a stack. Each stack contains a temperature sensor to detect temperatures in the stack. The memory capacity of a core storage device is determined by the number of memory planes in the memory stack. By varying the number of X and Y drive lines in each array (64 by 64, 128 by 128, and so on) and the number of memory planes in a memory stack, great flexibility can be achieved in the design of a core storage device's memory word length and number of addressable memory words. Stacks are usually divided into an **upper** and **lower** configuration for address selection and bit storage.

Three-wire magnetic core matrices allow for greater numbers of cores because less wiring is required. A single three-wire memory plane can provide 9 bits of storage over 32K addresses. Three-wire memory planes (core modules) are known as large plane memory.

Core Address Selection

The address selection process in core memory requires some unique circuitry. Because of the complex design of core memory, the address register and the translator use selectors and drivers to select the correct memory address. The address register bits are used to translate the bits to make the following address word bit selections:

- X and Y secondary selection
- X and Y primary selection

- X and Y diode selection
- Stack selection
- Inhibit half stack upper and lower

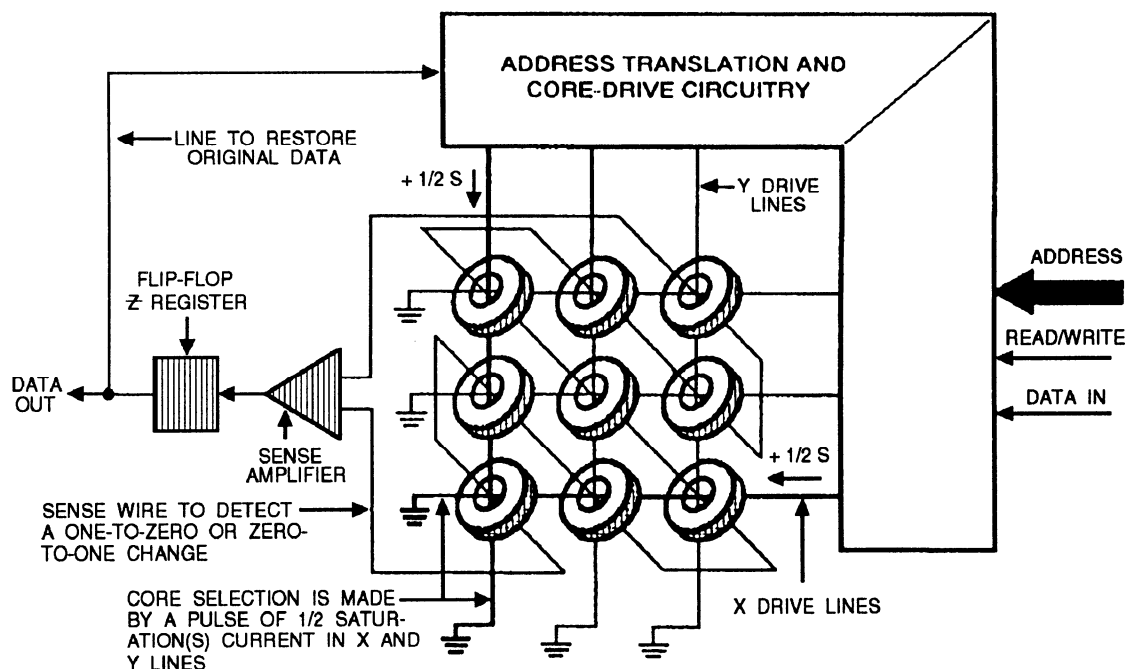
The circuitry associated with the address word bit selection includes the following:

- X and Y secondary selectors—provide enables for the secondary X and Y drive circuitry
- X and Y primary selectors—provide enables for the primary X and Y drive circuitry
- X and Y read/write diode selectors—enable the X and Y read drive line or the X and Y write drive line for one of four quadrants in all four of the memory stacks
- Inhibit selectors—activated only when writing zeros

Core Storage Read/Write Cycle

A cycle of events takes place whenever data is stored in or retrieved from core. Let's take a look at how data is read from core memory and then written or restored back into core memory.

READ CYCLE.— To determine the state of a core, **read current** is applied to both drive lines (X and Y or digit and word) passing through the core or cores addressed. Read current is designed to change the state



35NVM021

Figure 6-13.—Portion of a core memory plane in a stack.

of the core(s) to zero. If the core(s) had been in the one state, the sense line would pick up the change in flux direction from one to zero and indicate that a one had been stored in the core. At the completion of the read operation, the core or cores addressed are left in the zero state (three or four wire). The contents of the cores sensed by the read current would be temporarily stored in a register. This process of reading cores to the zero state is known as a **destructive readout**. Data read from cores must be temporarily saved in a register and then immediately written back into the cores (**restored**). The destructive readout of data from cores necessitates the completion of the storage or memory cycle.

WRITE (RESTORE) CYCLE.— New data or the data read from core must be written (or written back) into the cores for permanent storage. This portion of the storage cycle is known as the **write** or **restore** operation. Each storage cycle consists of a read and then a write or restore operation. Other terms commonly used are the read and write **half-cycles**. During the write or restore operation, **write current** is applied to the drive lines of the core or cores addressed. Write current is designed to change the state of a core from zero to one. Remember at this time all the selected cores are in the zero state from the read operation.

For the four-wire cores that are to store zeros, inhibit current is applied through the inhibit line in opposition to the write current. The inhibit current prevents the changing of the core(s) from zero to one.

Three-wire cores change to the ONE state only if both digit and word lines carry write current. Current in the digit line is dependent on the binary bit to be stored. The word line carries write current during the write operation. When the bit to be stored is a zero, no write current is applied to the digit line. Write current in the word line only will not change the state of the core to one. When a one is to be stored, write

current is applied to the digit line, and when combined with the word line current, changes the state of the core to one.

During the write or restore operation, all selected cores that are to store a one are written to the one state. All selected cores that are to store a zero will be inhibited from writing a one by the inhibit current in the four-wire cores and left in the zero state. Lack of digit write current will leave three-wire cores in the zero state.

READ/WRITE CIRCUITS.— Two important circuits used during the read/write cycle include the inhibit current regulator and sense amplifiers.

Inhibit Current Regulator.— The inhibit current regulator circuits are enabled only during a write cycle when there is zero in the corresponding bit of the interface register. The resulting inhibit current pulse prevents a one from being written into the associated bit position at the address selected.

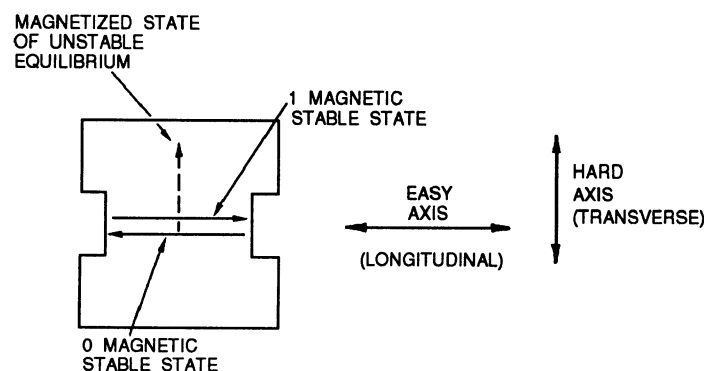
Sense Amplifiers.— The sense amplifiers sense the state of the cores selected during a read operation. The sense amplifiers are disabled during a write cycle, and enable the sense amplifier **stroke** during a read cycle. Data read by the sense amplifiers is transferred into the interface register until it can be restored back into core memory.

FILM MEMORY

Film memory is the other primary storage medium of digital computers. It is used primarily on large mainframes. Depending on the mainframe, film memory is contained in **memory modules**; usually four large memory modules to a mainframe computer set.

Film Architecture

Magnetic film storage is composed of hundreds of thousands of very small “I”-shaped magnetic thin film spots. Figure 6-14 is an example of one film spot and



35NVM022

Figure 6-14.—Bistable nature in thin film spots.

its bistable nature. Two paired thin film spots are used for each bit position. Like magnetic core memory, mated film memory uses bistable magnetic material as a means of recording binary data. A film spot is magnetized by current flow through the **word line** or **sense/digit line**. A film spot magnetized in one direction represents a binary zero, and when magnetized in the opposite direction represents a binary one. Figure 6-14 shows the bistable nature in thin film spots. As stated, a film spot is magnetized by current flow through the **conductor** (word line or sense/digit line). The preferred direction of magnetization is known as the **easy (longitudinal) axis** of the film because dipole alignment along this axis is stable. The axis perpendicular to the easy axis is referred to as **hard (transverse) axis** because dipole alignment along this axis is unstable and will fall to a stable state upon removal of the polarizing magnetic field.

The method of switching states in a mated film memory cell is referred to as coherent rotation. In **coherent rotation**, each magnetic dipole is rotated in unison with an applied field. A thin film can be switched as fast as external fields can be applied and removed. Figure 6-15, frame A, shows a film in the quiescent state, the magnetization is along the easy axis, and no external fields are applied. In frame B of figure 6-15, an external field is applied perpendicular to the easy axis. If the transverse field (H_T) is large enough, the magnetic vector will rotate to the hard axis position. As illustrated in figure 6-15, frame C, a small longitudinal field (H_L) is applied in the same direction as the desired zero or one easy axis, by applying current in the proper direction to the sense/digit line. Combining the H_L field with the H_T causes the magnetic vector to rotate beyond the hard axis to the desired polarization for a zero or a one. When the H_L and H_T fields are removed, the magnetic field falls or rotates (fig. 6-15, frame D) to the stable one state along the easy axis.

In figure 6-16, the method for applying an external field to the thin film elements is shown. Transverse fields are produced by passing current down the word line. Longitudinal fields are produced by passing current in the proper direction along the sense/digit line. When a current is passed through either conductor, a magnetic field is induced around the conductors as shown. Current through the word line will apply a transverse field to each element. The transverse field is concentrated on each film spot by a **magnetic keeper**. Current through the sense/digit line applies a longitudinal field in one of two directions to each film element. Each film element is physically separated

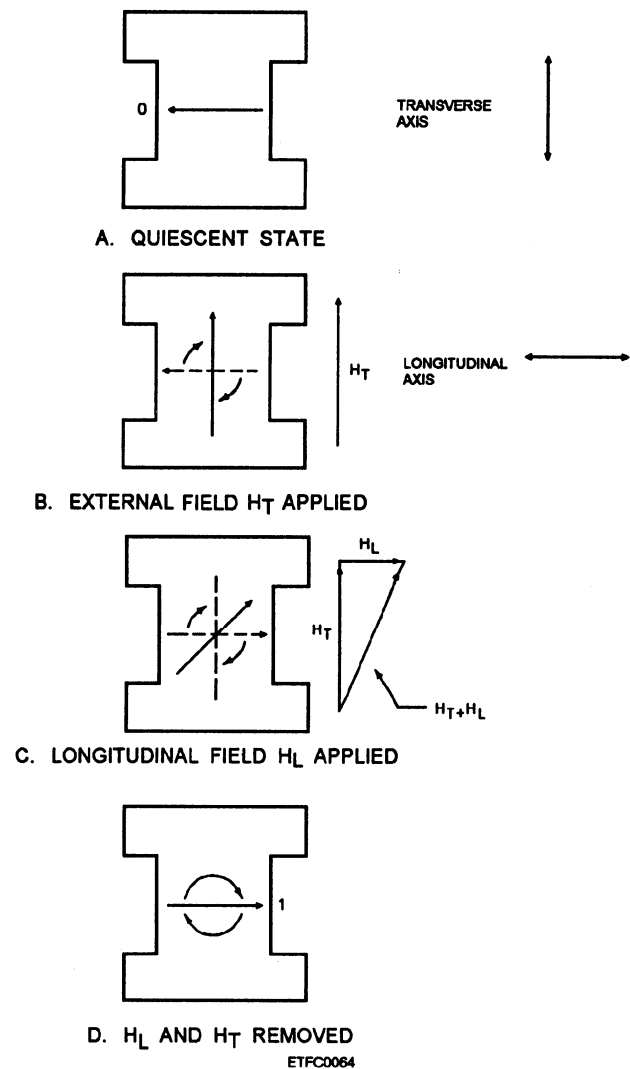
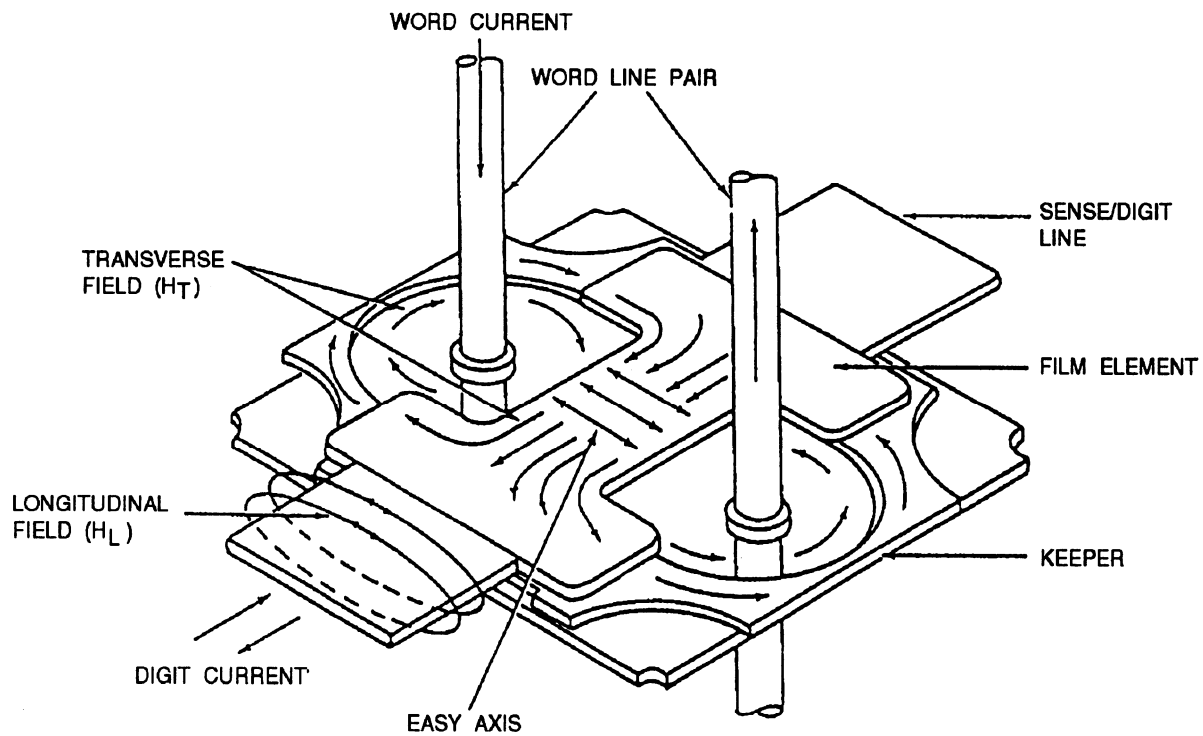


Figure 6-15.—Film architecture: A. Quiescent state; B. External field H_T applied; C. Longitudinal field H_L applied; D. H_L and H_T removed.

from the sense/digit line by a thin layer of electrical insulation.

PACKETS.— Film spots are organized by **packets**; the packets form a **stack**. Each packet can store two binary bits (zero or one) of data. Two paired thin film spots are used for each bit position. The purpose of the second film is to act as a **keeper** to the first film. This makes the mated film cells less susceptible to the disturbance from other cells in close proximity to them. The word pair line, sense/digit line, and keeper through each packet allow for the reading and writing of data in film. A packet consists of the following:



35NVM023

Figure 6-16.—Method for applying an external field to thin film elements.

- Word line pair
- Sense/digit line
- Film array
- Ground plane
- Keeper
- Insulators

An example of one packet is shown in figure 6-17.

Word Line Pair.— A current generated along the word line produces a **transverse field**. This magnetizes the film element. This causes the magnetic field to align with the word line and causes a current in the sense/digit line. This resulting current in the sense/digit line is read as a zero or a one by the register at the end of the sense/digit line.

Sense/Digit Line.— A current generated along the sense/digit line produces a longitudinal field in one of two directions to each film element. The longitudinal field is required in addition to the transverse field to assure proper writing into a thin film memory. The direction of the cell induced film signal on the sense/digit line determines whether a one or a zero will be written into memory.

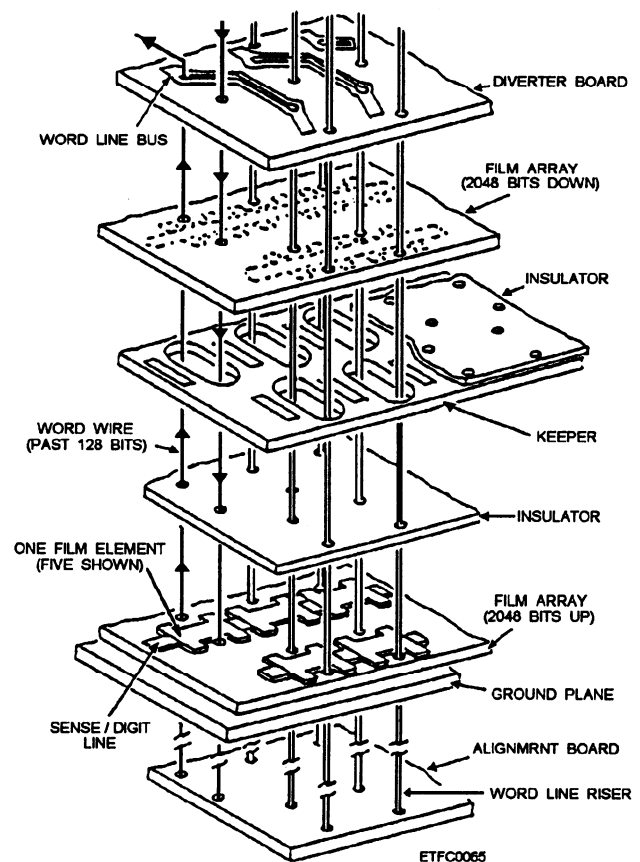


Figure 6-17.—One thin film packet.

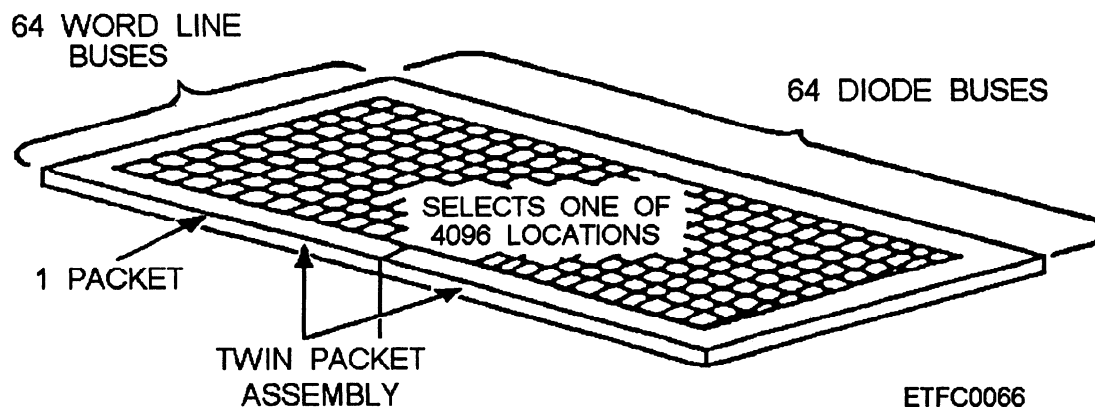


Figure 6-18.—Example portion of a film array.

FILM STORAGE LAYOUT.— Each paired thin film can store one binary bit of data. But because of their compactness, they have twice the storage capacity in the same volume as core memory. Mated film storage or film memory is designed to store a fixed number of **memory words**. Mated film memory is word organized. Each word line in a word organized memory is selected by a unique memory address. The film spots along a given word line are all the bits of a particular word. When a current is propagated through a selected word line, all the thin film spots along that line are read in a simultaneous parallel manner, each having a unique sense/digit line. Just like core memory, the size of film storage or its **memory capacity** is determined by the number of memory words that can be used or addressed to store and retrieve data. To accommodate a memory capacity of any size, the memory words are organized into **matrices**.

Matrices.— The mated film spots are organized into matrices called film arrays to simplify addressing, reading, and writing operations. Figure 6-18 is an

example portion of a film array. Each film spot in the film array has a word line pair and a sense/digit line; they affect the read/write operations. Mated film memories, like core memory, use a matrix.

Arrays.— In our example, a 64 by 64 array contains 64 word line buses and 64 diode buses. They select the exact word at a memory address location. By selecting one word line and one diode, one of the 4096 ($64 \times 64 = 4096$) memory locations in the array will be selected. One 64 by 64 array forms two packets of a **memory stack**.

Memory Stack.— In mated film memory, the packet is the building block of the film memory stack. The mated film memory stack contains all the film spots of the device, the associated circuitry that includes the word lines, sense/digit lines, and associated hardware. The associated hardware includes diode stick assemblies and boards (diverter, insulators, and ground planes, and alignment). Refer to figure 6-19 as an illustration of a memory stack.

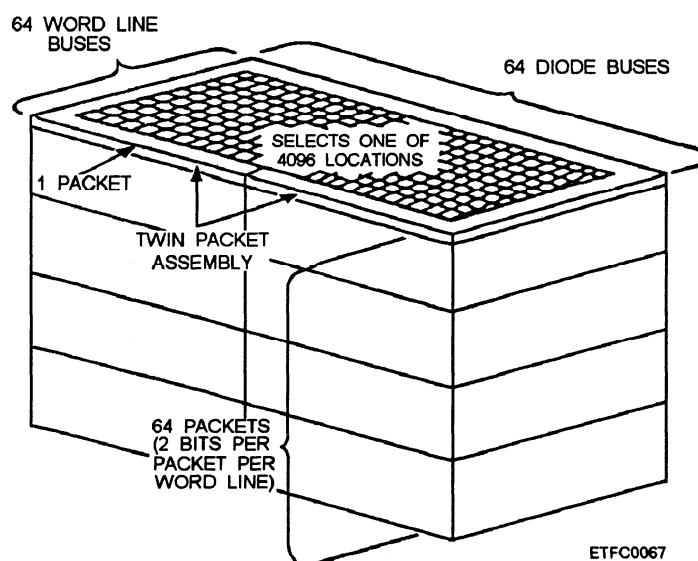


Figure 6-19.—Illustration of a memory stack.

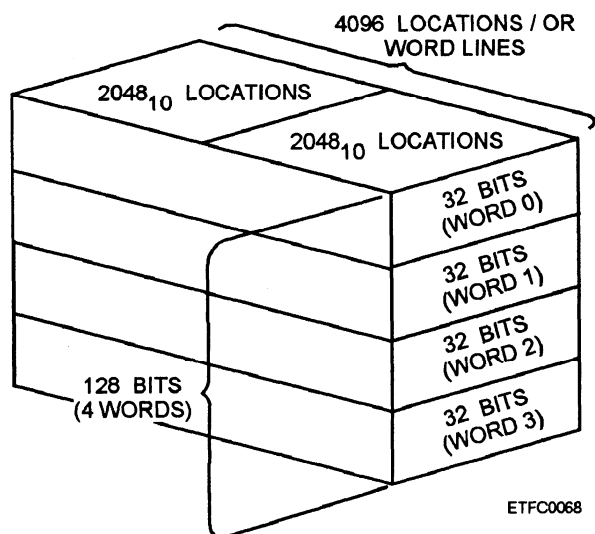


Figure 6-20.—Memory location words.

The memory capacity of a film core storage device is determined by the number of packets and the size of the arrays in the memory stack. By varying the number of word lines and diode assemblies in each array (64 by 64, 128 by 128, and so on) and the number of packets in a memory stack, great flexibility can be achieved in the design of a mated film storage device's memory word length and number of addressable memory words. Mated film stacks are usually divided

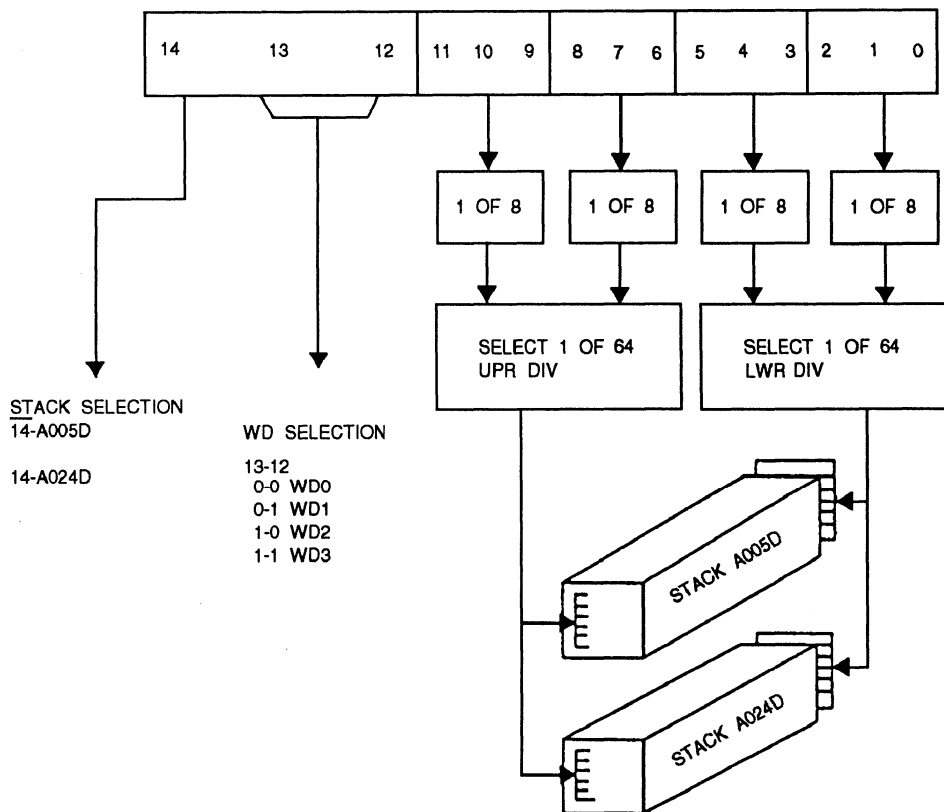
into words at each memory location (fig. 6-20). Instead of one word of so many bits in length at a given location, the computer has the option of selecting one of up to four words at each memory location.

Mated Film Address Selection

The address selection process in mated film memory uses the row (x)/column (y) concept just as core memory does. With mated film memory, the ultimate goal is not only to select an address location but to select a word at that memory address location. The upper bits in the address register are used to select the stack and the word at the memory location. The remainder of the bits are used to form matrices that in turn select one of an upper or lower diverter; this circuitry will select the location of the memory address. The address register bits are used to translate the bits to make the selections in the following order:

- Stack
- Location (if a 64 by 64 array, one of 4096)
- Word at the address location

Figure 6-21 is an example of the register used to select a memory word at an address location.



35NVM024

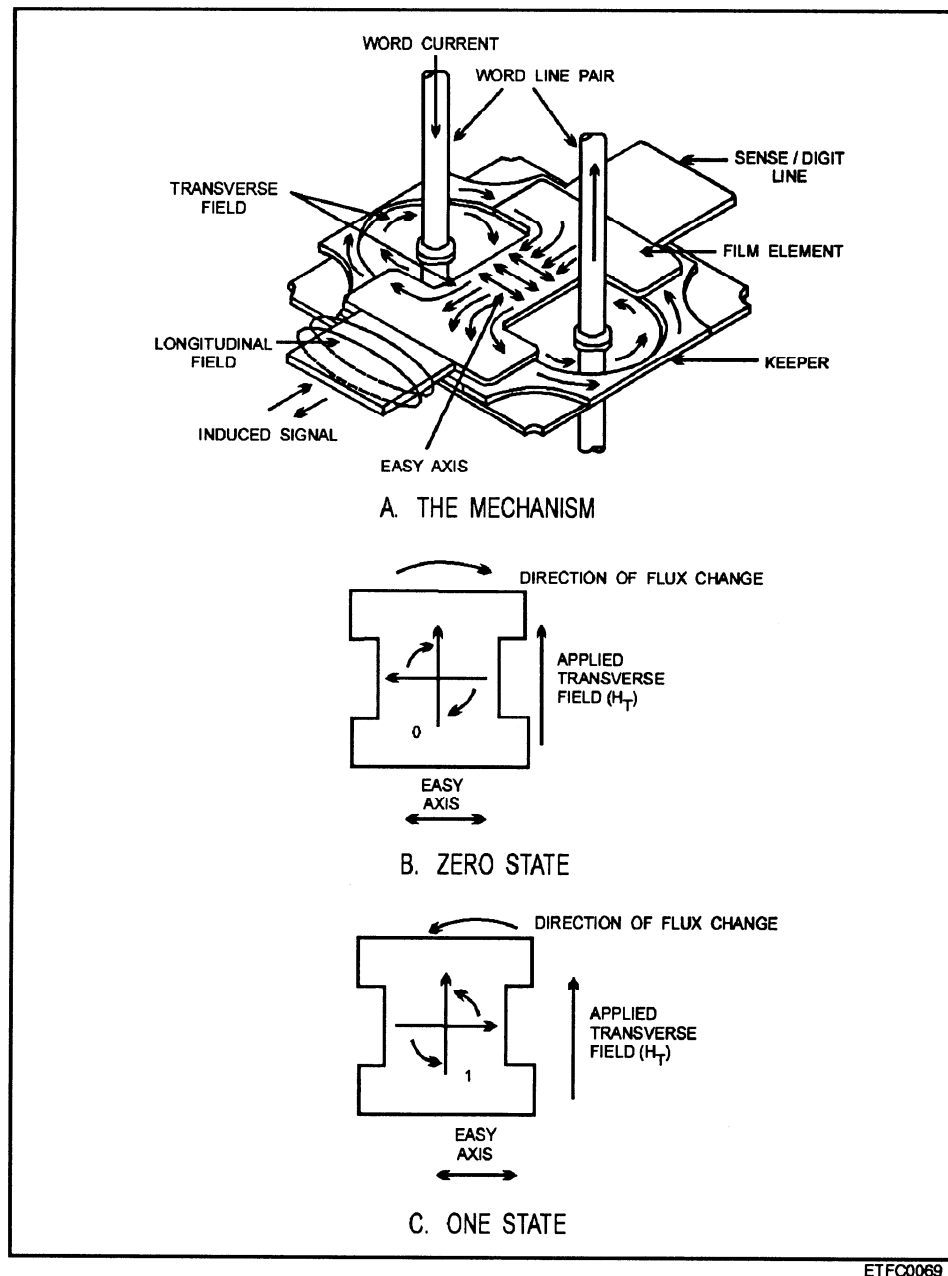
Figure 6-21.—Address register used to select location and word.

Mated Film Storage Read/Write Cycle

Similar to core memory, mated film memory must restore data back into memory after it has been read from memory so it will not be lost.

READ CYCLE.— To read a mated film memory cell, a current is generated along the word line and a transverse field is applied to the thin film cell. The rotation of the magnetic vector when it aligns with the word line, causes a current in the sense/digit line. This resulting current in the sense/digit line is read as a binary zero or one by the register at the end of the sense/digit line. This mechanism is graphically shown in frame A of figure 6-22.

The mated film memory cell in frame B of figure 6-22 is in the zero magnetic state. When the word field is applied, the magnetic polarization vector is rotated 90° to the hard direction. The clockwise direction of flux change induces a small voltage generating current in the sense/digit line as shown. The thin film in frame C of figure 6-22 is in the one state. The transverse field is applied by driving a pulse down the word line and the vector is forced in the hard direction, but now the flux change is counterclockwise. This flux change also induces a small voltage generating current in the sense/digit line, but it is the opposite polarity of the signal read from the film, thus storing a zero.



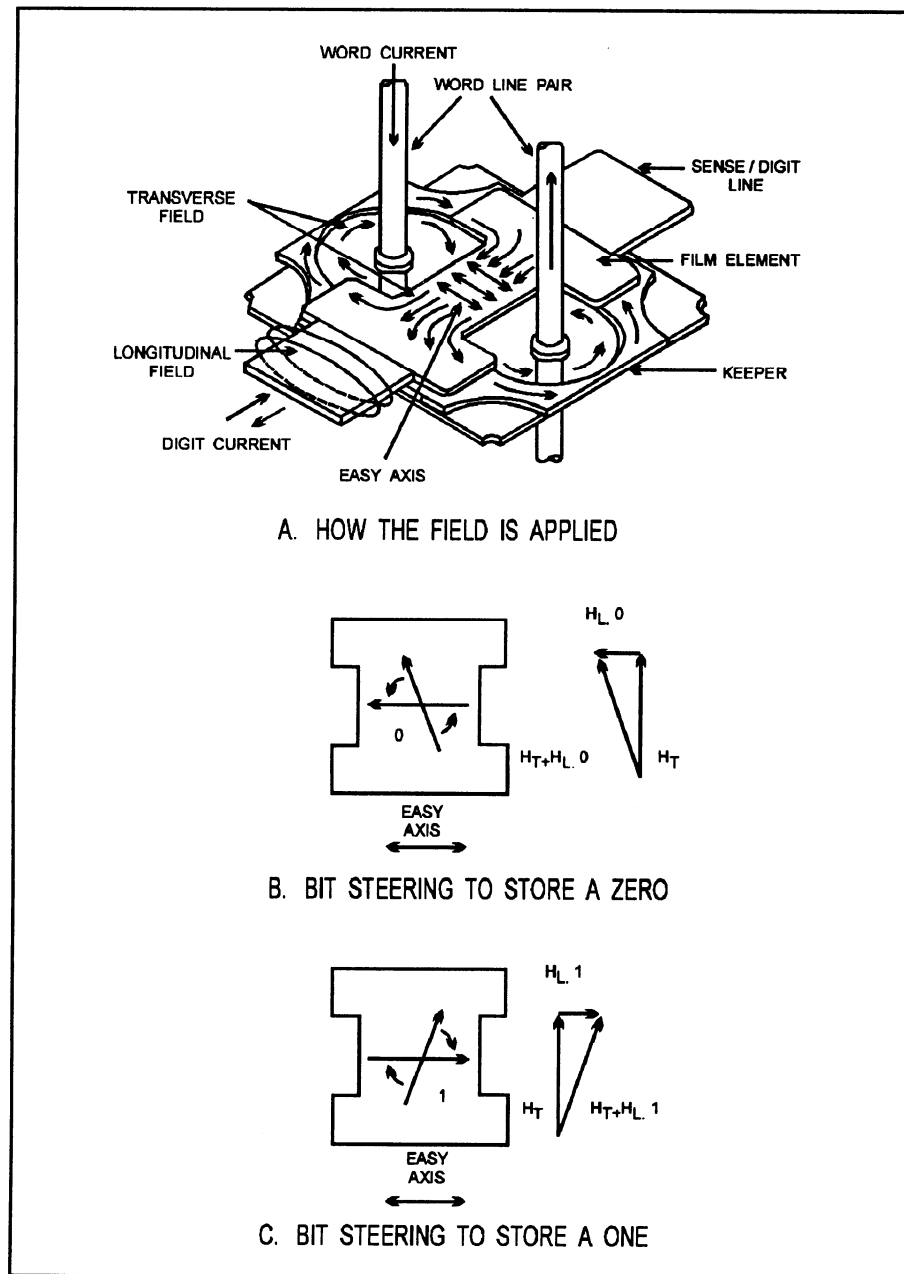
ETFC0069

Figure 6-22.—Reading a zero or a one: A. The mechanism; B. Zero state; C. One state.

The direction of the cell vector rotation induced film signal on the sense/digit line will determine what was the recorded state of the film. As the film is read, the cell magnetic state vector is forced in the hard direction. If the transverse field is removed, the film would fall back to one of the two easy states, a one or a zero. The actual state that a film would return to, however, would be uncertain, and a small amount of demagnetization of the film may occur. For this reason, reading a film in this manner is considered **destructive readout**. To ensure the film returns to its original state, the computer has an automatic hardwired restore operation, which is the same as writing into film

memory. This operation is an internal operation and is not controlled by the software user.

WRITE (RESTORE) CYCLE.— Film memory is like core memory; the data read from film must be written back into the film for permanent storage. This portion of the storage cycle is known as the **write** or **restore** operation. Each storage cycle consists of a read and a write or restore operation. In writing information to a film spot, a longitudinal field is required in addition to the transverse field to assure proper writing. Frame A of figure 6-23 shows how this field is applied. This longitudinal field is applied by passing a current



ETFC0070

Figure 6-23.—Writing (restoring) a zero or a one: A. How the field is applied; B. Bit steering to store a zero; C. Bit steering to store a one.

through the sense/digit line generating a magnetic field that is perpendicular to the transverse field. The direction of the current pulse generated up and down this line and the resulting longitudinal field are determined by whether a zero or a one is to be written.

Frame B of figure 6-23 illustrates the bit steering to store a zero. For a write operation the film's magnetic vector is first put in the hard direction from the transverse field. A small longitudinal field is then applied bypassing a current through the sense/digit line in the proper direction. This longitudinal field steers the vector toward the zero state. The word current is then removed, which further accelerates the magnetization towards the zero state. Then the digit current itself is removed leaving the film in the zero state. Frame C of figure 6-23 shows that when a one is to be stored, the bit current in the sense/digit line is reversed in direction from that used to store a zero. The resulting longitudinal field now steers the vector to the one state.

READ/WRITE CYCLE CIRCUITS.— Three very important circuits used in mated film memories during the read/write (restore) cycles are as follows:

- **Word current generator** —The word current generator produces the current pulse that provides the switching field for the memory film elements. The current generator is also used to produce drive pulses used for strobing during the memory cycle.

- **Digit drivers** —The digit drivers supply pulses required to write or restore data during the write/restore portion of all full memory cycles. Input to the drivers is supplied by the output of the data register's flip-flops. With these signals, the digit drivers are enabled to generate drive pulses, which write a logic high or a logic low in the address bit location in accordance with the binary (one or zero) contents of the data register. The binary value determines the direction of the digit current on the sense/digit line.

- **Sense amplifiers** —similar to core memory, the sense amplifiers sense the state of the data contained in the film element for storage in a data register for transmission of the data word or restoration of a one or a zero. Selected bits in the address register determine which group of bits are transmitted as a data word or changed by a memory write cycle.

SEMICONDUCTOR MEMORY

Semiconductor random access memory, or RAM, as it is often referred to, is used in all types of computers. RAM is also called a *read/write memory* or a

scratch-pad memory. Semiconductor RAM refers to semiconductor IC memories that can be used in a read mode as well as a write mode. Semiconductor memories use either a **read cycle** or a **write cycle** depending on the type of request, independent of each other. The read cycle is normally a shorter time period than the write cycle.

Semiconductor memories are normally **non-destructive readout** and **volatile** memories. In a nondestructive readout memory, the data stored in memory is not destroyed by the procedure used to read the data from the memory cells. Volatile memories require electrical power to maintain storage. If the power goes away for some reason, the data stored in the memory cells is lost. For this reason, an **uninterruptable power supply (UPS)** and a **battery backup system** are used in many semiconductor memory applications to maintain constant power and prevent loss of information because of power fluctuations or failures. This is especially important in microcomputers where configuration data is maintained in special devices such as a complementary metal-oxide semiconductor (CMOS). The battery backup and a filter capacitor provide the required power when the microcomputer has been powered down. Computers that use an UPS system have an established time in which data will be retained for momentary power losses.

The term random access memory (RAM) is consistently used for read/write devices. Although RAM only describes one characteristic of read/write devices, it is used and understood by most people to mean read/write devices. RAM means random addresses can be presented to the memory which means data can be written and read in any desired order from any location. Note: The term RAM is not used for *read-only memories* (ROM), although a ROM can also be random access.

Let's explore the basic building block of semiconductor memories: the RAM chip. Then we discuss the two main types of semiconductor RAM memories: static RAM (SRAM) and dynamic (DRAM), and variations of the two. SRAMS are faster but require more logic than DRAMS; thus they are more expensive than DRAMS.

The RAM Chip

In semiconductor memories, the basic building block is the RAM chip (fig. 6-24). This is true whether the memories are static or dynamic memories and are pcb's in a memory module or a pcb or pcb's mounted singularly. The semiconductor RAM itself is made up

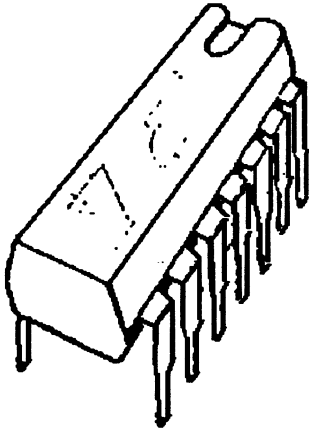
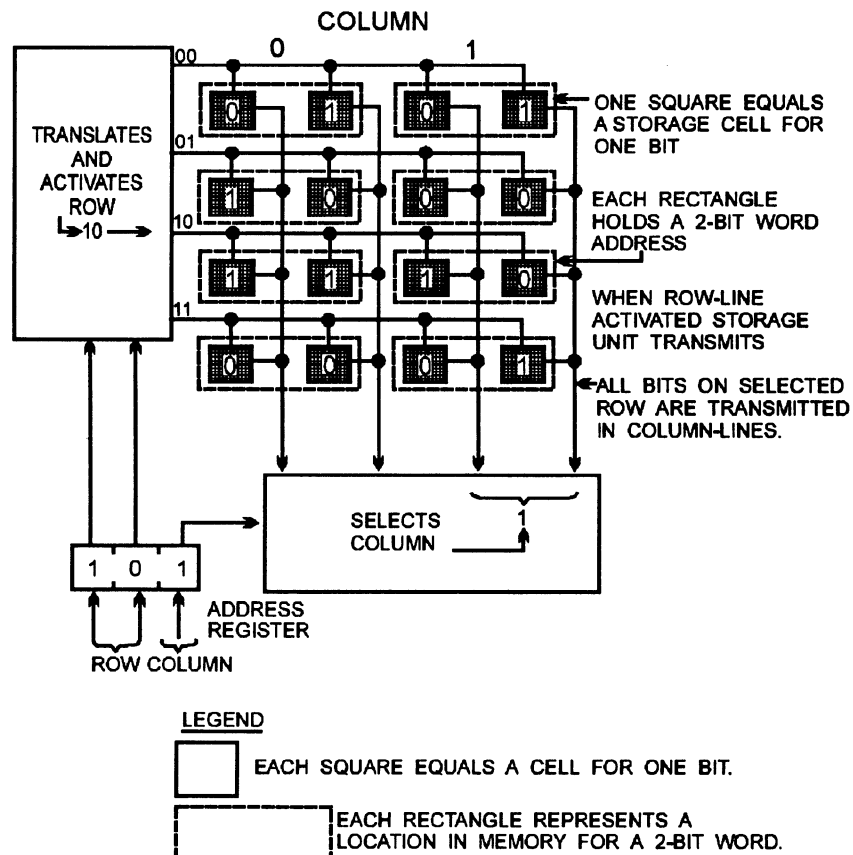


Figure 6-24.—RAM chip.

of variable numbers of these RAM chips. Each chip contains large numbers of memory cells and the logic to support them. Each memory cell is an electronic circuit with at least two stable states. With the advent of large and very large scale integration (LSI/VLSI), literally thousands or hundreds of thousands of memory cell circuits can be placed on a single chip. Each of the two-state memory cell circuits is capable of storing a single binary digit (0 or 1).

Figure 6-25 shows the general idea of how one-bit storage units (or cells) of any type are typically arranged so that stored information can be read out at random. The same arrangement works for writing data into a RAM. Notice the row and column arrangement; this is the same concept used by magnetic read/write memories. As a simple explanation, look at the memory shown in figure 6-25. It stores only 16 bits, as eight words of two bits each; notice the row-and-column arrangement. These chips are mounted on logic boards or circuit card assemblies (pcb's) in some sort of memory array, also called **gate arrays**, based on the memory capabilities required or desired by the equipment designer.

The capabilities of individual chips determine the array organization for the memory capabilities desired. On RAM chips, memory cells are organized based on two factors, the **number of memory words or addresses** and the **number of bits per word**. Most memory logic chips are rated by these values. For instance, a 4K by 16 chip would provide 4,096 16-bit memory addresses. This 4K by 16 chip will not support a 32-bit word for 4,096 addresses. The best it can do is the lower or upper half of the word. To support a 32-bit



ETFC0071

Figure 6-25.—One-bit storage units.

word, it would take two 4K by 16 chips to provide 4,096 addresses of 32 bits each.

To illustrate the random access nature of RAM, the number of words or addresses, and bits per word, we offer a simple illustration. Figure 6-26 shows the organization of a 64-bit memory. The 64 squares (mostly blank) in the figure represent the 64 positions that can be filled with data. Notice that the 64 bits are organized into 16 groups called **words**. Each word contains four bits of information. This memory is said to be organized as a 16×4 memory. That is, it contains 16 words, and each word is 4 bits long. The total number and capabilities (16 by 4 and so forth) of these individual circuits will define the total memory capacities of the respective computer.

In our example, the total number of memory cells is 64. There are many variations in the ways a 64-bit memory could be organized: 64×1 , 32×2 , or 8×8 . The memory in figure 6-26 looks very much like a truth table on a scratch pad. On the table after word 3, you'll notice the contents of word 3 is (0110). We say we have stored, or written, a word into the memory; this is the write operation. To look at the contents of word 3, we simply read the contents of word 3 using the read operation. What is also important about RAM memory is that we can read or write into any word on the table and in any order, that is why it is called *random access*.

ADDRESS	BIT 3	BIT 2	BIT 1	BIT 0
WORD 0				
WORD 1				
WORD 2				
WORD 3	0	1	1	0
WORD 4				
WORD 5				
WORD 6				
WORD 7				
WORD 8				
WORD 9				
WORD 10				
WORD 11				
WORD 12				
WORD 13				
WORD 14				
WORD 15				

Figure 6-26.—Organization of a 64-bit memory.

Static RAM (SRAM)

Static random access memories (SRAMs) are semiconductor integrated circuits that use a **flip-flop** application for each storage cell. Figure 6-27 illustrates

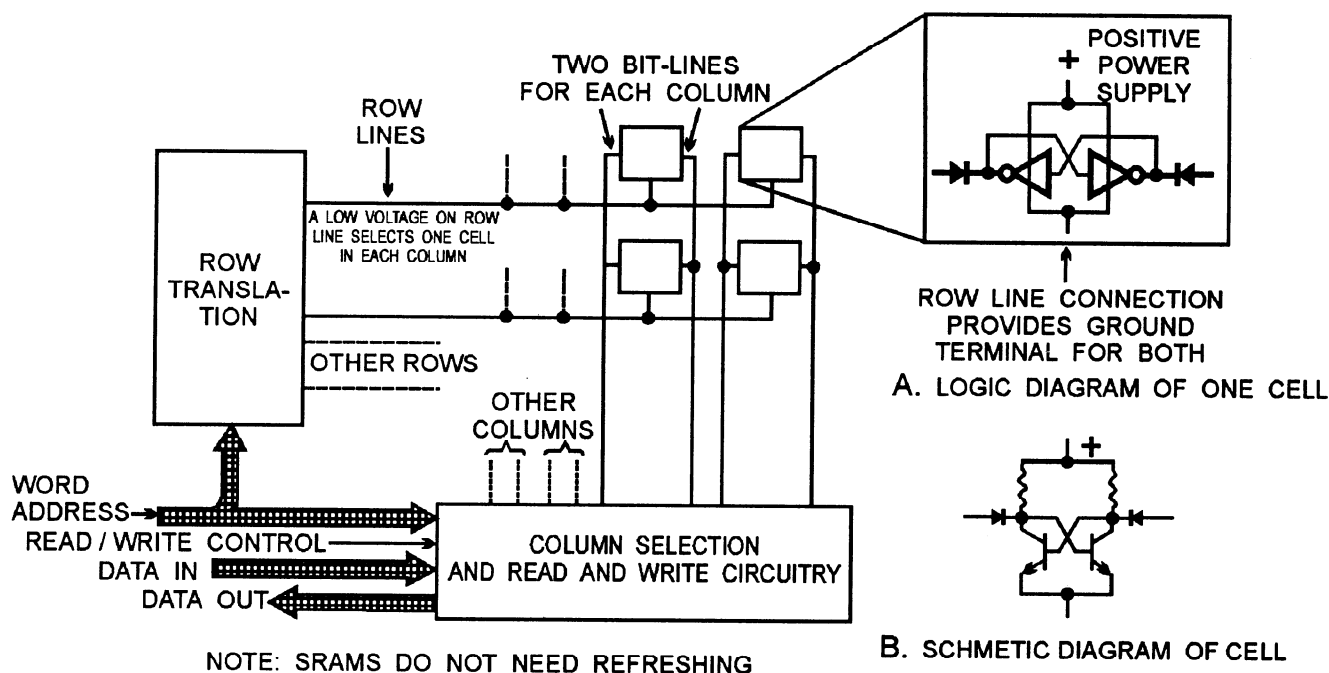


Figure 6-27.—SRAM cell and associated circuitry.

a static RAM cell and its associated circuitry in block form. Each memory cell can latch, or store, data in a stable state. Information is written into and readout of the cell through the column lines. The characteristics of flip-flops keep the flip-flop in its present state and allow you to read the data out of the cell without changing its state when the row-line is activated. Similarly data is written through the column line only when the row-line is activated, so only one cell in each column is selected. A read/write control signal controls reading and writing operations. The zero or one state in the cells can be held indefinitely as long as proper power supply levels are maintained.

D-type and **R-S type** flip-flops are commonly used for SRAMs. The flip-flops can be made of either bipolar or MOS transistors. MOS yields a higher density but lower access speed. Bipolar RAMs have a higher access speed but take up more space. Figure 6-28, frames A and B, and figure 6-29 illustrate schematic diagrams of individual bipolar and MOS RAM cells. Figure 6-28, frame A, is a diode-coupled

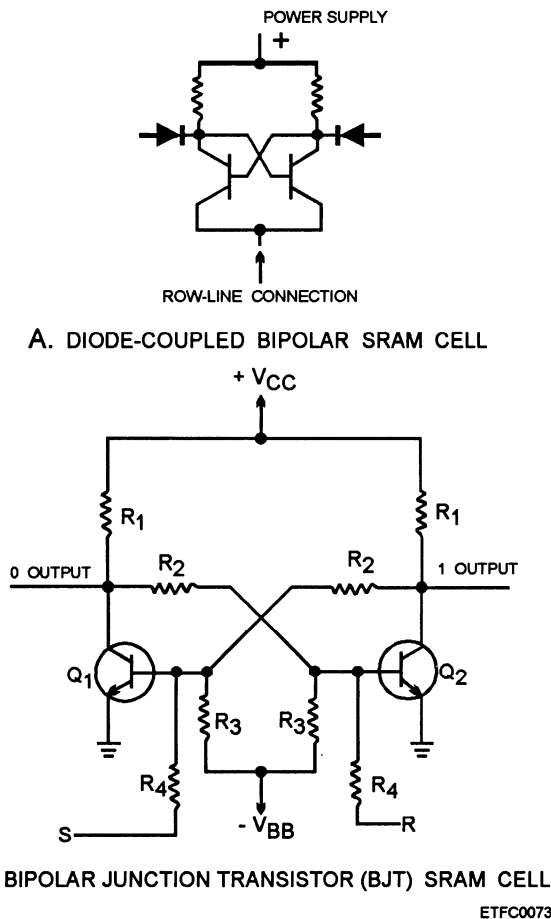


Figure 6-28.—Examples of SRAMs: A. Diode-coupled bipolar SRAM cell; B. Bipolar junction transistor (BJT) SRAM cell.

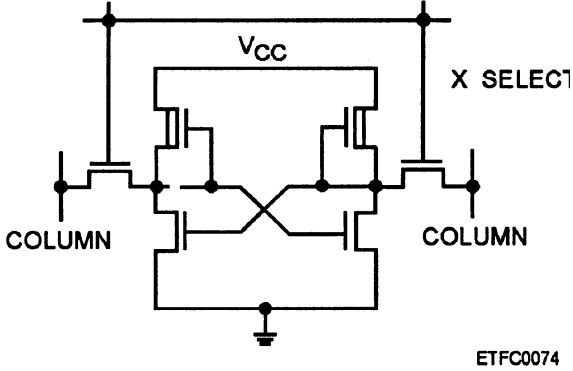


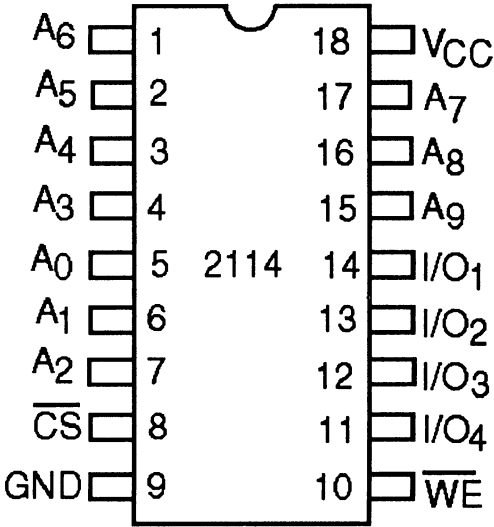
Figure 6-29.—SRAM MOS cell.

bipolar static RAM cell; figure 6-28, frame B, is bipolar junction transistor (BJT) static RAM cell; and figure 6-29 is a static RAM MOS cell. As stated, the RAM chip is mounted in a logic array on a pcb. Figure 6-30 is an illustration of an IC chip, with pin connections used in a static bipolar or MOS RAM.

RAM chips come in various configurations and sizes. The number of IC RAM chips needed for a computer's RAM memory is determined by the requirements and memory size of the computer. Let's use an example IC to discuss the operation of a RAM chip, which includes the architecture, address selection, and read/write cycles.

Static RAM Organization and Operation

Our example RAM uses a 1 K by 4 configuration, 1024 words that are 4 bits in length. Many groups of 1K by 4 RAM chips can be grouped together with simple support logic to form larger memory systems. A



35NVM026

Figure 6-30.—SRAM IC chip with connections.

block diagram of a 1K by 4 is shown in figure 6-31. This RAM chip uses 10 address lines to address 1024 words. The address is provided over 10 input lines (A_0 through A_9). The 10-bit address is internally translated within the chip. Bits A_6 through A_9 feed the column select circuits while bits A_0 through A_5 feed the row select circuits. The address lines are used to enable the addressed memory cell flip-flop circuits by row and column number. The 10 address lines form 2^{10} , or 1024, possible conditions. Each addressed word has 4 bits.

There are 4,096 memory cell flip-flop circuits in a 64 row by 64 column memory array. Within the 64 by 64 memory array, only 4 flip-flops enabled by both row and column signals can be set or cleared by the data bits during a write operation or can have their outputs sensed during a read operation.

Data is stored in, or read from, the memory cells via the four I/O lines, I/O_1 through I/O_4 . To provide stable signals within the memory cell array, the four I/O lines are buffered, as shown -on the block diagram. The address lines are usually tied to the computer or memory system address bus, while the I/O data lines are tied to the data bus. The I/O lines are bidirectional. For write operations, they carry the data to be written into the memory cells. For read operations, they carry the output of the memory cells.

The remaining pin connections shown on the block diagram are used for control and power. The **chip select** (\overline{CS}) line is an input used to enable a particular part, or group of parts, out of a large memory array. For

example, a 16K by 8 memory can be from 32 of the 1K by 4 static RAM chips. Only two chips are selected during any single read or write sequence. The chip select (\overline{CS}) signal, when true, indicates that the particular chip's circuitry has been selected for a read or write operation. The chip select signal originates from a higher level decoder circuit, which controls several RAM chips. Common address and data lines connect all the chips under the decoder but only the chip receiving the chip select will handle the data.

The **write enable** (\overline{WE}) input line is used to determine whether a read or write operation is taking place. The write enable signal is generated from the computer system. When the chip select is active, a write pulse on the write enable line is used to store data within the memory cell array. The internal circuitry of the chip will accept data from the I/O lines and set or clear the selected row and column flip-flops to match the bits on the I/O lines. The data buffers are switched to input mode during a write cycle. During a read cycle, the write enable is false indicating that the read cycle is being processed and the data buffers are switched to the output mode. During a read operation, the internal cell data is output to the computer data bus. The contents of the flip-flops themselves are not changed by the read operation.

The V_{cc} and ground lines are used to supply power to the memory IC. Power consumption varies slightly with the mode of operation of the static RAM. Atypical 1K by 4 static RAM uses 5 volts of dc power, and typical power consumption is 500 mW.

Dynamic RAM (DRAM)

Dynamic random access memories (DRAMs) are semiconductor integrated circuits (ICs) that operate like a bank of capacitors. DRAMs consist of MOS transistors. Figure 6-32 is an example illustration of a dynamic RAM cell and its associated circuitry. The cells are capacitor-type circuits; a charged cell equals a logic 1, while a discharged cell equals a logic 0. Each cell consists of a MOS transistor and a tiny capacitor. When a row-line is activated, all the MOS transistors on that row are turned on, connecting their capacitors to their column lines. By way of the column lines, the capacitors are charged when writing and the charges are detected when reading. Electric charges are put into the cells through the column lines and read out through the same lines, using appropriate switching circuitry in the column selector section. Words from the data input in figure 6-32 are written into the capacitors through the column lines and the data is readout through the same

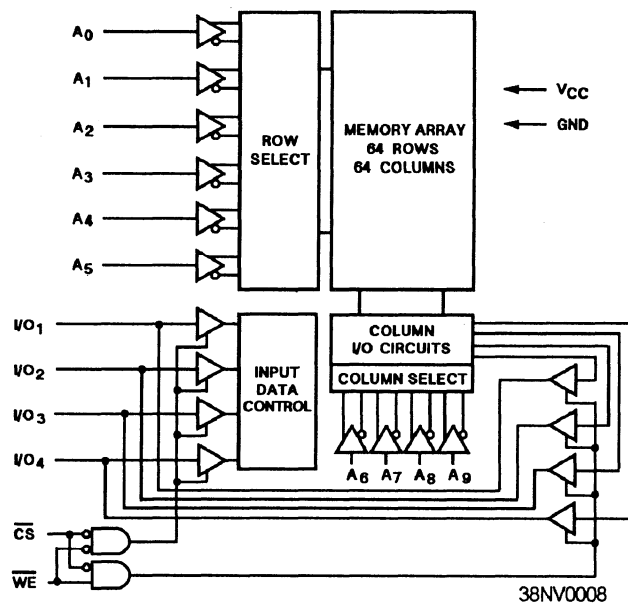


Figure 6-31.—Block diagram of a 1K × 4 SRAM.

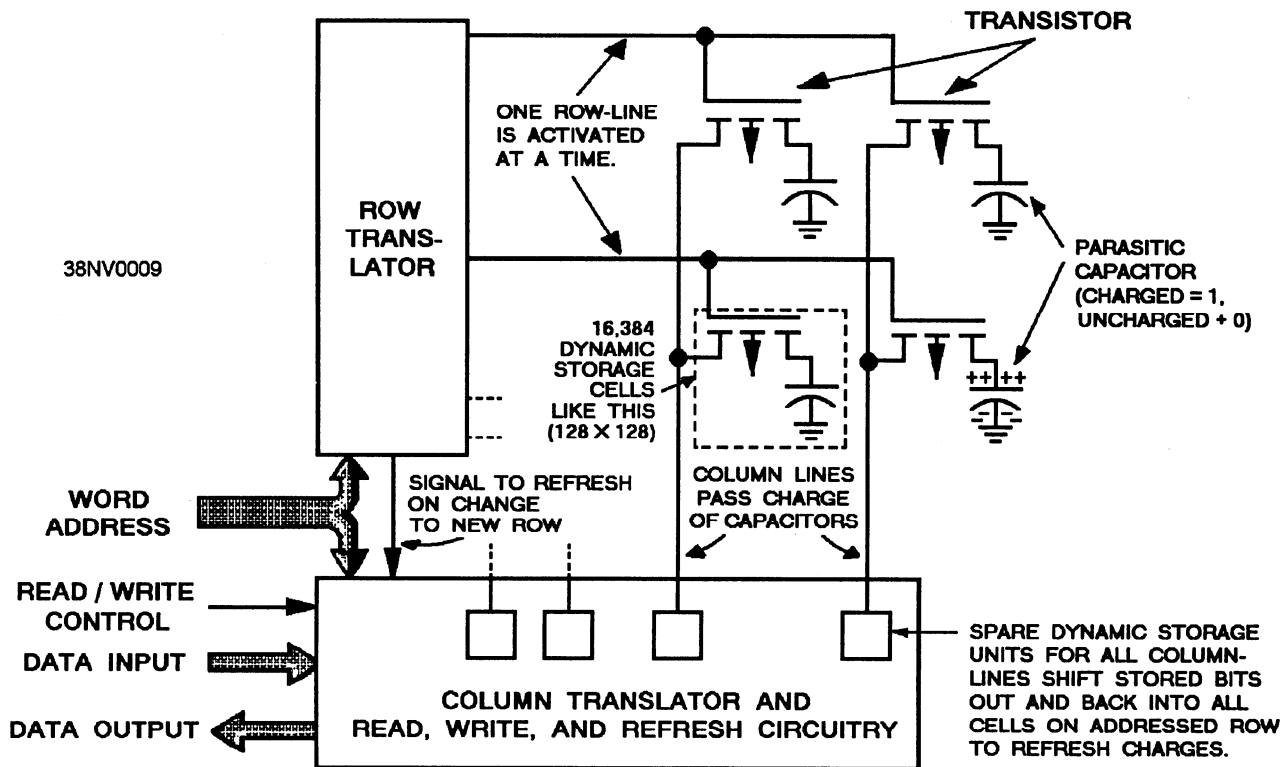


Figure 6-32.—DRAM storage cell.

column lines using switching circuitry in the **column selector** section. A read/write control tells memory whether to read or write.

The dynamic RAM cell is less complex than a static cell because it does not use a latch to store data. A parasitic capacitors formed in the integrated circuit and this becomes the storage element, as pictured in figure 6-32. The single transistor switch is used to isolate or select one particular cell from the entire memory array. Because the basic dynamic cell design is simple and contains few elements, it is possible to achieve much higher densities than with static cell designs.

A typical memory system may be formed from many 16K word dynamic RAM parts. These parts are usually structured as 16K by 1, or 16,384 words of only one bit each. Larger memory words are made by ganging as many chips as required. This allows greater flexibility for system designers to organize memory systems with a small or wide data bus (for example, 16K by 8, 16K by 16, or 16K by 32). Circuit boards that are extremely cost-effective for large mainframes are easily created with densely packed dynamic RAM parts.

Power consumption by dynamic RAM is another advantage over static RAM. Because the dynamic RAM cell does not use a latched design to hold data as

do the static RAM parts, the power consumed by each cell is negligible when it is not being accessed. Most of the power consumed occurs during a read or write operation and a small amount is consumed during the refresh cycle. The lower power consumption of dynamic RAM leads to lower cooling requirements and smaller power supplies because of the reduced memory power needs.

One disadvantage that a dynamic RAM has is the need to **refresh** the entire memory array within a certain period of time (usually two milliseconds). The DRAM memory cells do not retain their charged state for more than a few milliseconds. DRAM cells are subject to degradation due primarily to time and temperature. To retain information, the content of each memory cell must be **refreshed** before the charge dissipates. The memory array may be refreshed in one of two ways: **externally** or **internally**. Externally is more cost effective because a single refresh address generator is shared by the entire memory array. A type of DRAM, time volatile memory (TVM) uses a battery backup to maintain refresh for 30 minutes after the computer has been powered down or power has been lost.

In our explanation of dynamic RAM organization and operation, we use two examples to discuss the architecture, address selection, and read/write/refresh

cycles. One uses an external refresh and the other an internal refresh.

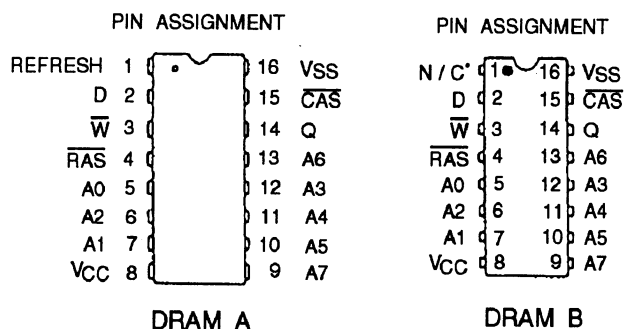
Dynamic RAM Organization and Operation

Our example dynamic RAM chips both have 64K of memory. We label them dynamic RAMs **A** and **B**. Dynamic RAM **A** contains a built-in refresh circuitry, which is driven by a simple external clock, while dynamic RAM **B** must be refreshed by external logic. A pin-assignment diagram for these DRAMs **A** and **B** is shown in figure 6-33.

These dynamic RAM parts contain 65,536 1-bit words and require a 16-bit address word ($2^{16} = 65,536$). The address word is formed by a multiplex technique; whereas two 8-bit words are input in two steps from the eight address lines labeled A0 through A7. This 8-bit word must be formed by external logic that interfaces the computer memory bus to the memory system.

The D line is the data input line. The Q line is the data output line. These lines may be tied together or separated; it varies with the system. The D and Q lines are tied together in applications that call for a bidirectional data bus. However, separated D and Q lines speed up the system. In larger memory systems, all the dynamic RAM parts in the memory array share the address bus. The data bus is separated into individual data bits. Each bit is associated with one RAM Chip.

For timing and control, the system uses the refresh address strobe ($\overline{\text{RAS}}$) and the column address strobe ($\overline{\text{CAS}}$) lines. To signify when a write operation is being performed, the system uses a low level on the $\overline{\text{W}}$. The V_{cc} pin is used for 5 V power input. The V_{ss} is held at ground.



35NVM027

Figure 6-33.—DRAMs with pin assignments: A. DRAMA with built-in refresh circuitry; B. DRAM B which requires external refresh logic.

On DRAMA, pin 1 is used for refresh. The pin 1 refresh technique uses an internal 8-bit counter to generate the required 128 refresh addresses. Use of this pin requires a low-state clock pulse on the refresh line, while the RAS signal is sent to a high state. The refresh clock increments the refresh address with each clock pulse. With external logic, this technique is fairly inexpensive. The main disadvantage of this IC is the additional internal refresh logic.

The alternate refresh technique can be used on both DRAMs A and B. This alternate technique uses the $\overline{\text{RAS}}$ and $\overline{\text{CAS}}$ lines to control the refresh mode. The $\overline{\text{RAS}}$ line is sent low, while the $\overline{\text{CAS}}$ line is sent high, and the refresh address is presented from external logic to the DRAM memory array. All 128 refresh addresses must be presented within two milliseconds, as is the case for the self-refresh mode.

TOPIC 3—READ-ONLY MEMORY (ROM)

In modern computers, portions of the available main memory addresses and special local memories are made up of read-only memory (ROM). ROMs are used for various memory applications, such as fixed program storage, look-up tables, and code conversions. The programs on the ROM are actually more hardware than software; therefore, they are often referred to as **firmware**. ROM has all the operational characteristics of read/write memories except that data cannot be written into the ROM addresses by the normal computer accessing methods (write request). You can only read and/or execute the contents of each ROM memory address. The contents of the ROM addresses can be used over and over again without alteration, and the data does not have to be written back into ROM. The primary use of a ROM allows the computer to perform its I/O operations, which is one of the primary functions of the CPU. In this chapter, we discuss how ROM works and the different types of ROMs; how and when they are programmed. The term *non-destructive readout (NDRO) memory* is often used to describe ROM used in militarized computers. Newer computers use ROM and the different variations of ROM in the NDRO to store the bootstrap and other special-purpose programs.

READ-ONLY MEMORY (ROM) ARCHITECTURE

ROM is consistent in all computers. Remember it is tailored to meet each computer's needs. ROM comes

in various sizes: from 512 to 8K words. The size will depend on the computer and the functions of its ROM. The arrangement of ROM uses the same concept as main memory. A row (x)/column (y) arrangement is used to obtain the ROM addresses. The array of the ROM cell matrix also varies with the size of the ROM. Certain portions of main memory are set aside for the ROM addresses. When ROM is selected at the front panel or as part of a start up routine for basic input/output system (BIOS), the computer will default to the addresses established by the manufacturer. ROM comes in one of two basic packages: either a **module** or a **ROM chip**. To make changes to ROM programs, you

must remove the applicable ROM module or ROM chip(s) from the CPU module or the pcb on which they are mounted and replace it/them with the module or chip(s) containing the new version of the program.

ROM (NDRO) Modules

Nondestructive readout (NDRO) memories may consist of a single pcb or several pcb's that come in a single module (fig. 6-34). They are usually located in a larger CPU module or as part (one pcb) of a group of pcb's located in a chassis.

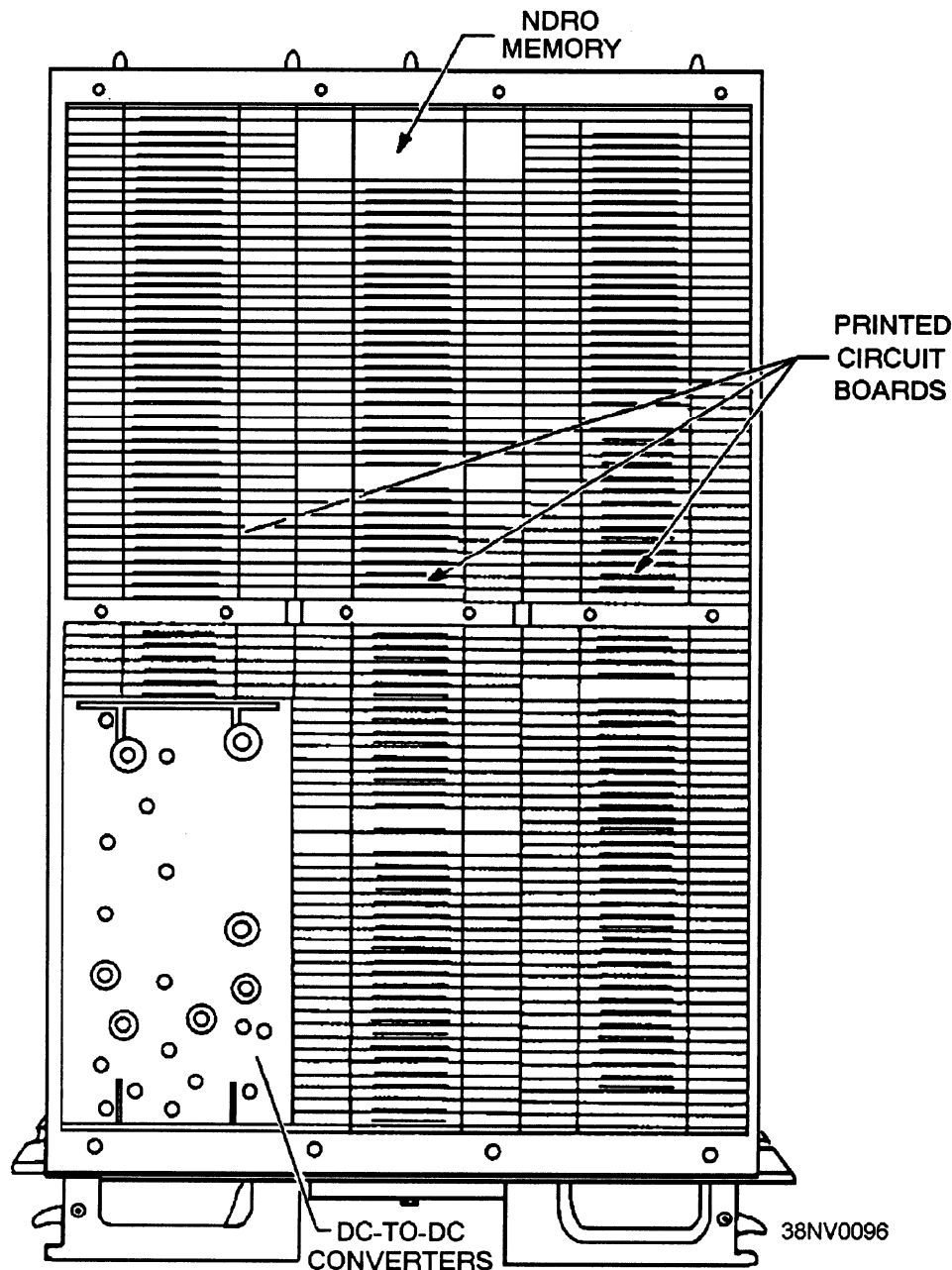


Figure 6-34.—Mainframe CPU with an NDRO memory identified.

ROM Chips

A ROM chip is one or more chips on a pcb located in a rack or backplane/motherboard. Figure 6-35 is an example of a 64K ROM block diagram and IC with pin connections. Notice the items used to obtain the ROM address. The heart of the chip is the 65,536-bit memory array. This array is masked with a ROM data pattern. The desired word in the array is selected by the X and Y decoders. The 13 address lines are the inputs to these two decoders. As the address is decoded, the output word is presented to the output buffers. The chip select line is used to enable or disable the tristate mode of the output buffers. The eight output lines come from the output buffers. Pure ROM chips are manufactured with the desired software instructions or data installed.

READ-ONLY MEMORY (ROM) MATERIALS

The types of materials that make up a ROM also vary. ROMs can consist of the following types of materials:

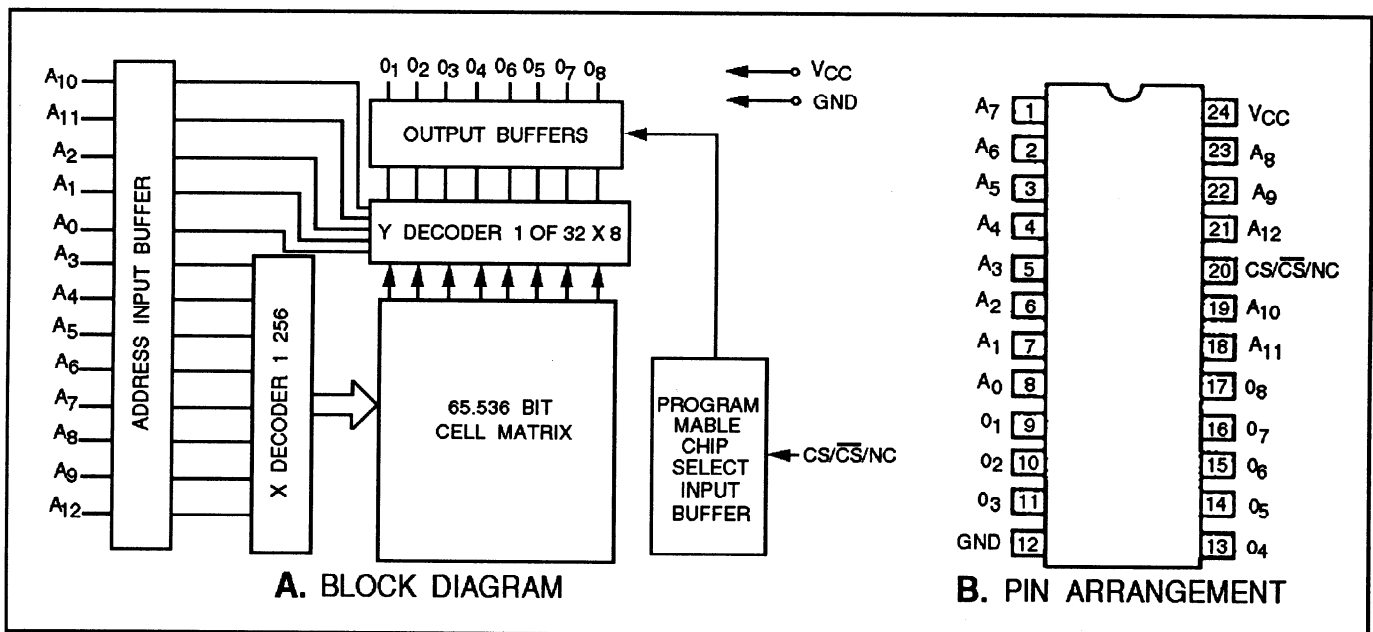
- Hardwired (fig. 6-36)

- Magnetic (fig. 6-36)
- Transistors-Bipolar or MOS (fig. 6-37; MOS ROM)
- Fusible links

Regardless of the type of material used for ROM, the cell array is masked to a particular 0/1 arrangement to form the permanent data needed for ROM operations.

READ-ONLY MEMORY (ROM) OPERATIONS

As stated, ROM operations are characteristic of main memory operations except you cannot write to ROM. This means ROM operations use no write pulses or equivalent and no data input buffers are needed. The basic operation of ROM, like a main memory address, is to use the permanent data stored in the ROM address. Since we are studying the computer and its internal operations, the operation in this case is the execution of an instruction contained at a ROM address. But you can see where if you needed to use the same data at a particular location repeatedly (for example, a look-up table), this concept could be used.



35NVM028

Figure 6-35.—ROM chip: A. Block diagram; B. Pin arrangement.

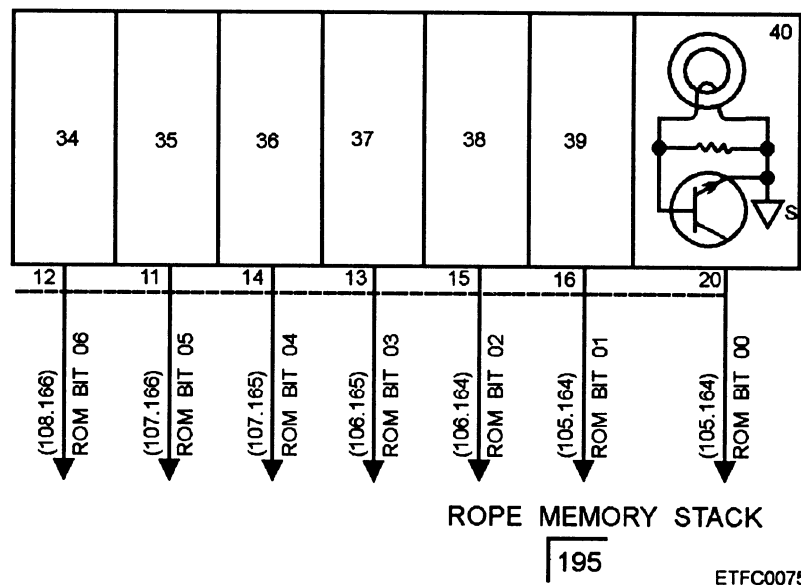


Figure 6-36.—Hardwired magnetic ROM.

For ROM operations to take place, the following events generally take place in the following order:

1. ROM is either selected on the computer's front panel or equivalent of a computer; or the computer is turned on and it is part of the start up routine.
2. The ROM address is translated.
3. The ROM address is selected.
4. The contents of the ROM address are sent to a designated register for transfer to the instruction register.
5. The instruction is sent to the instruction register for translation.
6. The instruction is executed.

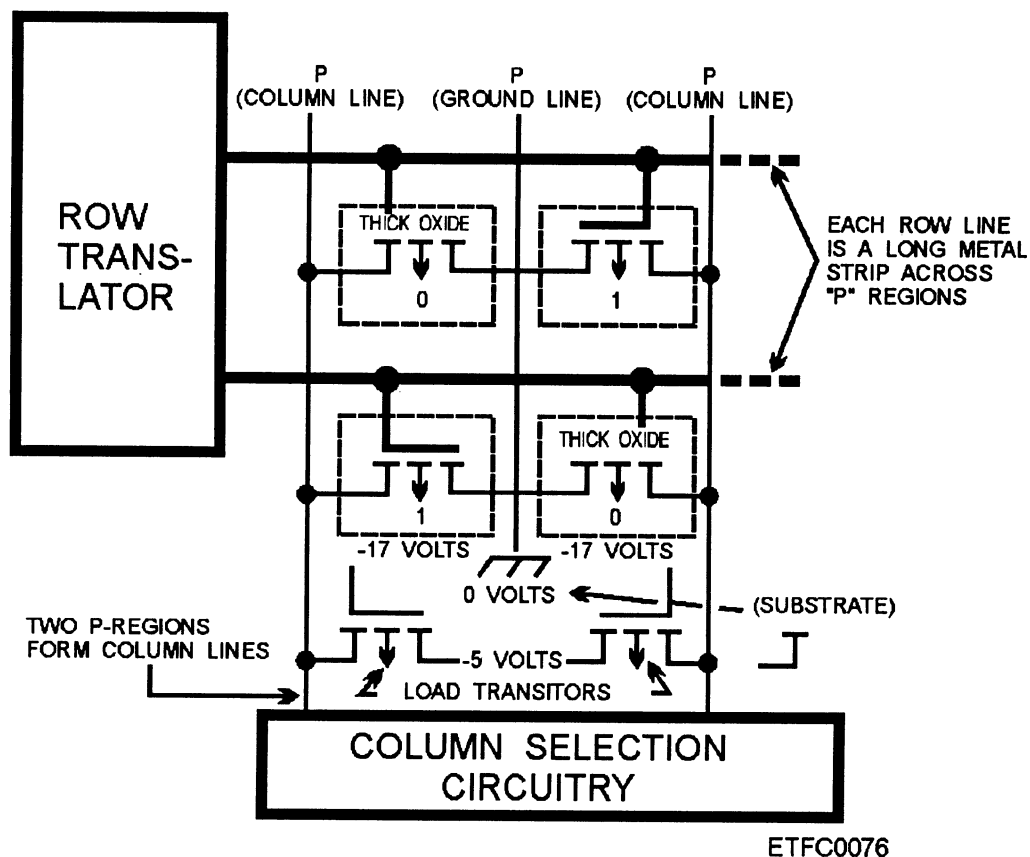


Figure 6-37.—MOS ROM.

Once ROM operations are completed, the computer is ready for normal operational use. For all this to take place, ROM uses circuits in the computer that we have already discussed. In some cases, they are circuits specific for ROM operations. They include:

- Registers and flip-flops
- Timing
- Control signals
- Internal bus

READ-ONLY MEMORY TYPES

Types of ROMs include the basic ROM that once manufactured cannot be written on again. Other types, called programmable read-only memories (PROMs), can be written on again and again.

Read-Only Memory

ROMs are prepared at the factory. They are not meant to be changed by the user or the technician. They are only to be changed when a newer version is authorized and supplied to replace the old one.

Programmable Read-Only memory (PROM)

A PROM is a programmable ROM. Once programmed it acts like a ROM. It can be field-programmed by an authorized technician. Each cell is identified by selecting the row and the column just like locating an address in read/write memory. There are two types of **PROMs—erasable** and **nonerasable**. Erasable PROMs can be erased and reprogrammed. Nonerasable PROMs **cannot** be changed once they are programmed.

There are a couple of ways to create or erase the ones in the array; **electrically** or with **ultraviolet (UV) light**. Some PROMs are electrically programmed but erased with UV light. Others are erased electrically and programmed with the UV light.

ELECTRICAL.— An electric charge can be used to either blow fusible links permanently in a cell or used on a special transistor with two gates. With the special transistors, the gate between the memory cell and the column wire is disabled by the electrical charge.

UV LIGHT.— UV light is used to erase data in a cell by exposing the IC die to the UV light for a few minutes (usually less than 30 minutes). UV light can

also be used to restore ones to a cell by dissipating the electrical charge that disabled the gate.

ELECTRICALLY ALTERABLE OR ERASABLE PROM (EAPROM OR EEPROM).— The EAPROM or EEPROM can be programmed (modified) or erased while it is still in the circuit and used like a nonvolatile read/write memory. EAPROMs/EEPROMs use an electric charge to erase the ones. Some types of EAPROMs/EEPROMs are more versatile; individual cells can be reprogrammed by reversing the voltage used to create a zero. There are some timing constraints that cause the part to need more time for erasure or programming than is needed to read data from the part. Some EAPROM/EEPROMs have a **word** or **byte** erase mode.

ULTRAVIOLET-ERASABLE PROM (UV EPROM OR EPROM).— UV EPROMs/EPROMs trap a charge (1) in the cells to represent the data. To release the charge, the cells are exposed to the UV light for 30 minutes or less. UV EPROMs/EPROMs are usually programmed out of circuit. Figure 6-38 is an example of a $2K \times 8$ UV EPROM; block diagram and pin assignments.

SUMMARY—COMPUTER MEMORIES

This chapter introduced you to memory types. The following information summarizes important points you should have learned.

MEMORY— The main memory of a computer is used for storing programs, data, calculations, and operands.

MEMORY MODULES— Memory modules are made up of multiple pcb's (support circuitry) and memory components (stacks [core or film] and semiconductor pcb's) to form one memory module or unit. Memory modules are interchangeable with other modules of the same type and size in the same computer set. Each module provides a fixed number of memory words with a fixed number of bit positions for each word.

MEMORY ARCHITECTURE— Memories are typically organized in square form so that they have an equal number of rows (x) and columns (y). Each intersection of a row and a column comprises a memory word address. Each memory address will contain a memory word.

MEMORY OPERATIONS— Memory operations operate on a request, selection, and initiate basis. A memory request or selection and a memory word

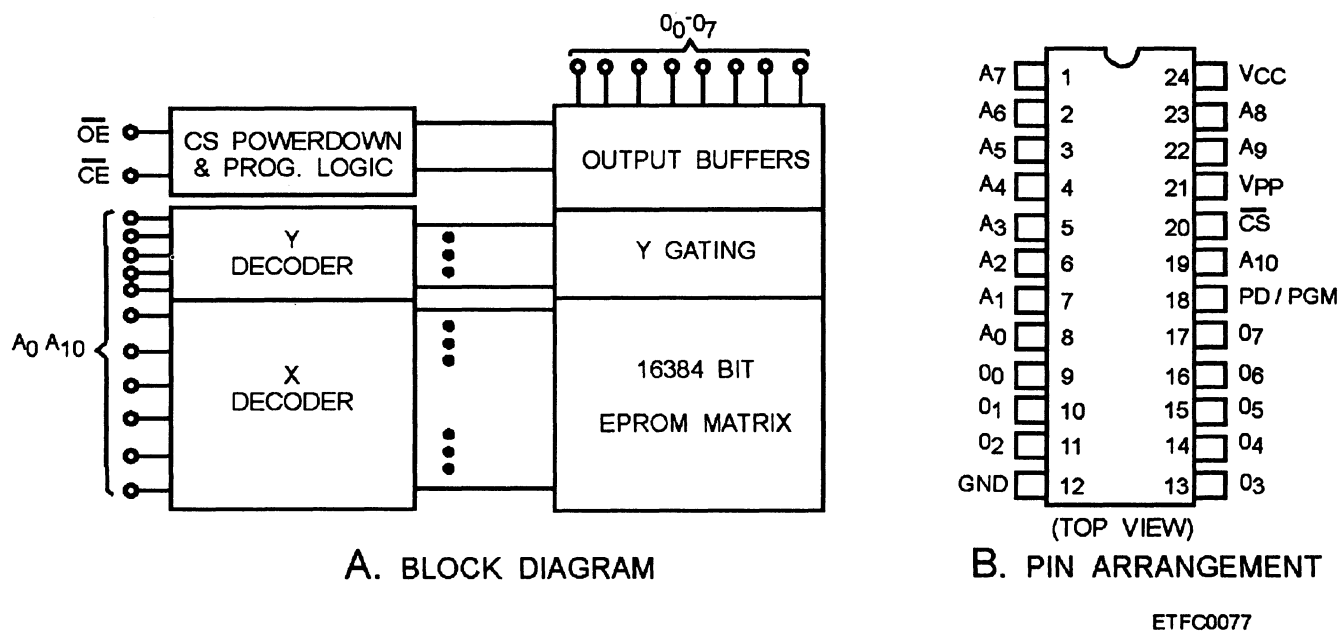


Figure 6-38.—Example of a MOS UV EPROM: A. Block diagram; B. Pin arrangement

location are transmitted from the requestor (CPU or I/O sections) to the memory section. The computer's internal bus system transmits the memory request or selection and location to the memory section.

READ/WRITE MEMORY— In read/write memories, the data can be retrieved from memory, altered, and written back into memory. Read/write memories are random access memories. They are categorized according to the materials they are constructed from and not their basic operation.

CORE MEMORY— Magnetic core storage is composed of hundreds of thousands of very small doughnut-shaped ferrite cores. The ferrite cores are strung together on grids of very thin wires known as core planes. Each core can store one binary bit (0 or 1) of data. A core is magnetized by current flow through the wires on which the core is strung. A core magnetized in one direction represents a binary zero, and when magnetized in the opposite direction, a binary one. The direction the core is magnetized is dependent on the direction of current flow through the wires on which it is strung.

FILM MEMORY— Magnetic film storage is composed of hundreds of thousands of very small "I"- shaped magnetic thin film spots. Two paired thin

film spots are used for each bit position. A film spot is magnetized by current flow through the word line or sense/digit line. A film spot magnetized in one direction represents a binary zero, and when magnetized in the opposite direction represents a binary one.

SEMICONDUCTOR MEMORY— Semiconductor RAM refers to semiconductor IC memories that can be used in a read mode as well as a write mode. Semiconductor memories are normally nondestructive readout and volatile memories.

RAM CHIP— RAM chips make up semiconductor RAM. They contain large numbers of memory cells and the logic to support them. Each memory cell is an electronic circuit that has a least two stable states. Each of the two-state memory cell circuits can store one bit (0 or 1).

STATIC RANDOM ACCESS MEMORY (SRAM)— Static random access memories (SRAMs) are semiconductor integrated circuits that use a flip-flop application for each storage cell. The flip-flops are made of either bipolar or MOS transistors.

DYNAMIC RANDOM ACCESS MEMORY (DRAM)— Dynamic random access memories (DRAMs) are semiconductor integrated circuits (ICs)

that operate like a bank of capacitors. The cells are capacitor type circuits; a charged cell equals a logic 1 while a discharged cell equals a logic 0. Each cell consists of a MOS transistor and a tiny capacitor.

READ-ONLY MEMORY (ROM)— ROMs are used for various memory applications, such as fixed program storage, look-up tables, and code conversions. The programs on the ROM are actually more hardware than software (firmware). The contents of the ROM

addresses can be used over and over again without alteration. The primary use of a ROM allows the computer to perform its I/O operations.

PROGRAMMABLE READ-ONLY MEMORY (PROM)— A PROM is a programmable ROM. Once programmed it acts like a ROM. It may be erasable or nonerasable.

Learn all you can about the memories used in the computers you maintain.